September/October 1989 £2.75

COMMODORE Disk User

FOR C64 AND C128 USERS

SCRAPBOOK 64
A must for collectors

Disk Dungeons

Which Word Processor

Video Titling

Using the Rasters

Games Reviews

ISSN 0953-0614

09

9 770953 061007

September/October 1989 £2.75

COMMODORE Disk User

FOR C64 AND C128 USERS

SCRAPBOOK 64
A must for collectors

Dungeons

Which Word Processor

Video Titling

Using the Rasters

Games Reviews

ISSN 0953-0614

09

9 770953 061007

DATEL ELECTRONICS

DATA RECORDER

- Quality Commodore compatible data recorder.
- Pause control, counter, etc.
- Suitable for 64/128.
- Send now for quick delivery.

ONLY £24.99

SPRINT 128

- Full feature Centronics Printer Interface.
- Connect your 64/128 to a range of full size Centronics Parallel Printers.
- Easy to use - supports Commodore Graphics Set.
- Onboard Microprocessor Chip means no power programmes to load.
- Works with most applications.
- No more to buy!

ONLY £29.99

EPROM ERASER

- Erases up to 4 Eproms at once.
- Small size with built-in timer.
- Even fits in your pocket!
- Comes complete - no more to buy.

ONLY £39.99

ROBOTEK 64

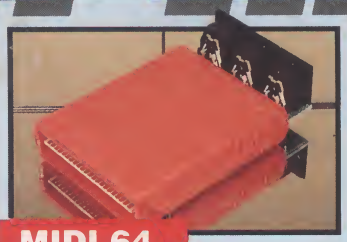
SPECIAL VALUE

MODEL & ROBOT CONTROL MADE EASY.

- 4 output channels - each with onboard relay.
- 4 input channels - each fully buffered TTL level sensing.
- Analogue input with full 8 bit conversion.
- Voice input for voice control.
- Software features:- test mode/ analogue measurement/voice activate/digital readout etc.

ONLY £39.99

INCLUDING HARDWARE/SOFTWARE/ MIC. ETC,(STATE TAPE OR DISK).



MIDI 64

FULL MIDI INTERFACE FOR THE 64/128 AT A REALISTIC PRICE.

- MIDI in/MIDI thru/2 x MIDI out.
- Compatible with most leading software packages.

ONLY £29.99

DIGITAL SOUND SAMPLER

- The new sampler allows you to record any sound digitally into memory & then replay it with astounding effects.
- Playback forwards/backwards with echo/reverb/ring modulation.
- Now with full sound editing module to produce outstanding effects.
- Full 8 bit D to A & ADC conversion.
- MIDI compatible with suitable interface. (i.e. Datel unit for £29.99, see ad.).
- Live effects menu includes real time display of waveforms.
- Line in/mic in/line out/feedback controls.
- Powerfull sequencer with editing features.
- Load/save sample. Up to 8 samples in memory at one time.
- Complete software/hardware package. Tape or Disk (please state).

ONLY £49.99

COM-DRUM

GITAL DRUM SYSTEM

- Now you can turn your digital sound sampler into a digital drum system.
- 8 digital drum sounds in memory at one time.
- Complete with 3 drum kits. Real drum sounds - not synthesised.
- Create superb drum rhythms with real & step time.
- Full editing. Menu driven. Load/ Save facilities.
- Output to hi-fi or through TV speaker.

ONLY £9.99

STATE TAPE OR DISK

COM-DRUM EDITOR

- 24 drum sounds supplied on disk to enable you to construct your own drum kit.
- Re-arrange sounds supplied with a Com-Drum to make new kits.
- Load/save facilities.

ONLY £4.99 **DISK ONLY**

PARALLEL PRINTER CABLE

- Connects full size printers to the parallel port of your C64/128.
- Many programmes and cartridges (Action Replay/Final Cartridge etc.) will drive printers from this port.

ONLY £12.99 **COMPLETE**

A TOTAL MIDI MUSIC PACKAGE

SAVE OVER £70

UNBEATABLE VALUE PACK - THE YAMAHA SHS 10 FM MIDI COMPATIBLE SHOULDER SYNTHESISER, THE ADVANCED MUSIC STUDIO SOFTWARE & MIDI INTERFACE PLUS FREE MIDI CABLES TO CONNECT EVERYTHING TO YOUR COMPUTER SYSTEM

FREE MIDI CABLES

YAMAHA SHS 10 FM SYNTHESISER KEYBOARD

- Superbly styled guitar-type keyboard with shoulder strap.
- Top quality brandname.
- 2.5 octave keyboard.
- 25 built-in instrument and rhythm choices.
- Uses FM synthesis.
- Full MIDI standard.
- Superbly styled guitar-type keyboard with shoulder strap.
- Requires 6 "C" batteries or AC/DC adapter.

ADVANCED MUSIC SYSTEM

- A full feature Music Studio with MIDI.
- Editing and sequencer features.
- Many features - too many to list!
- Sorry, disk only.

MIDI INTERFACE

- MIDI In/Out/Thru.
- Simply plug in.

FOR ONLY £99.99

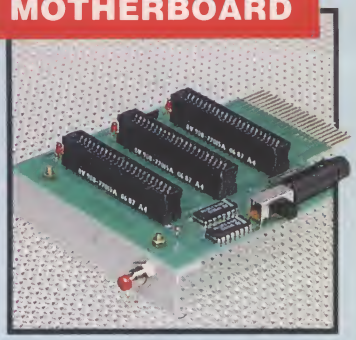
NO MORE TO BUY!!

RESET CARTRIDGE

- Unstoppable reset button. NOTE:- user port cheaper type reset buttons offered by others are not unstoppable.
- Resets even so called "unstoppable" programs.
- Add pokes from magazines etc.
- Simply plugs in to cartridge port.

ONLY £5.99

3 SLOT MOTHERBOARD



SAVE WEAR & TEAR ON YOUR EXPANSION PORT

- Will accept three cartridges on it's high grade PCB.
- Switch in/out any slot.
- Fully buffered.
- Reset button and an onboard safety fuse.

ONLY £16.99

TOOLKIT IV

THE ULTIMATE DISK TOOLKIT FOR THE 1540/1541.

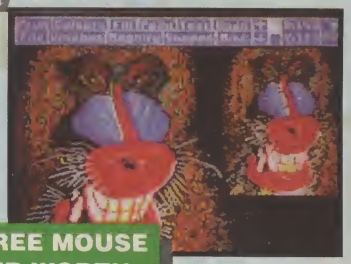
- A disk toolkit is an absolute must for the serious disk hacker. Toolkit IV has more features than most for less.
- DISC DOCTOR V2 - Read & write any track & sector including extra & renumbered tracks. Repair damaged sectors.
- HEADER/GAP EDITOR - Decodes & displays ALL header information including off bytes & header gap. Rewrite the entire header & header gap. Renumber sectors. Also edit any sector tail gap.
- DISK LOOK - Sort directory. Recover lost files. Display file start /end addresses. Disassemble any file program directly from the disk to SCREEN or PRINTER including undocumented opcodes. Edit Bam.
- FAST FILE COPY - Selective file copy. Works at up to 6 times normal speed.
- FAST DISK COPY - Copy an entire disk in 2 minutes or less using single 1541.
- FILE COMPACTOR - Can compact machine programs by up to 50%. Save disk space.
- FORMATTER - 10 second format an entire disk or format any individual track or half track 0 to 41.
- ERROR EDIT - Quickly find & recreate all read errors including extras & renumbered tracks or sectors & half tracks from 0 to 41.

ONLY £9.99

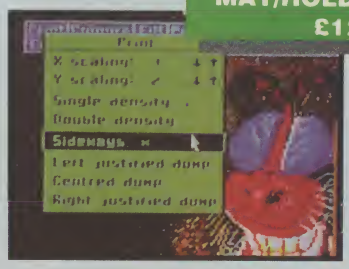
DATEL ELECTRONICS

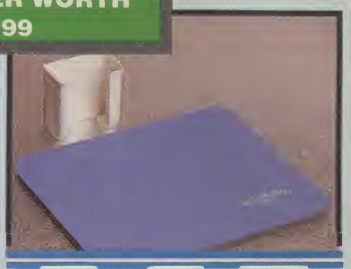
THE ULTIMATE GRAPHICS PACKAGE...

SPECIAL OFFER PACK!



INCLUDES FREE MOUSE MAT/HOLDER WORTH £12.99





COMMODORE 1351 MOUSE

COMPLETE WITH

OCP ADVANCED ART STUDIO™

- 1351 Mouse is a high resolution two button mouse featuring optical counting, teflon guides, microswitches, rubber coated ball and high quality interface.
- When combined with OCP Advanced Art Studio this graphics package is quite simply the best system available. The features are unmatched...

- Create an image - shrink it, expand it, move it, rotate it, copy it, colour it etc., etc.
- Spray patterns or shades, make elastic lines - stretch and manipulate shapes.
- Zoom in to add detail in fine mode.
- Pulldown/Icon driven menus for ease of use.
- Mouse operation, plus joystick and keyboard control.
- 16 pens, 8 sprays, 16 brushes - so flexible anyone can create superb graphics easily.
- Full cut and paste facilities plus excellent printer support.
- Pixel edit, font editor, flip, invert, rotate, solid or textured fill and professional manual make Art Studio simply the best graphics package for the Commodore (please state Tape or Disk).

ONLY £29.99

TOTAL PACKAGE INCLUDING 1351 MOUSE/ MOUSE MAT/ HOLDER AND OCP ART STUDIO

VIC 20 RAMPACK

- Simple plug-in memory expansion cartridges for your VIC 20.
- Never be short of memory again!
- Two models available - 16K or 32K.
- Both models are switchable for various memory settings i.e. 3K, 16K, 24K, etc.
- Many VIC programs require extra RAM - this is your answer.

ONLY £14.99 FOR 16K

ONLY £19.99 FOR 32K

EPROM ERASER

- This handy AC/DC Eprom Eraser will erase up to 4 chips quickly and easily.
- Built-in timer and safety switch on cover.
- Superbly styled - small enough to fit in pocket - even as it works!!
- Works from built-in battery (supplied) or from AC/DC adaptor if required.

ONLY £39.99

LIMITED NUMBERS AT THIS AMAZINGLY LOW PRICE!!

EPROMMER 64™

- A top quality, easy-to-use EPROM programer for the 64/128.
- Fully menu driven software/ hardware package makes programing/reading/verifying/ copying EPROMs simplicity itself.
- Will program 2716, 2764, 27128 & 27256 chips, 12.5, 21 or 25 volts.
- Fits into user port for maximum compatibility with cartridges/ Superom Board etc.
- Full feature system - all functions covered like device check/verify.
- We believe Eprommer 64 is the most comprehensive, most friendly & best value for money programer available for the 64/128.
- Ideal companion for Superom Board, Cartridge Development System, our kernal expanders or indeed any EPROM base project.
- Comes complete with instructions - plus the cartridge handbook.

ONLY £39.99

COMPLETE

NOW YOU CAN INCREASE THE MEMORY OF YOUR C16 EASILY

C16 RAMPACK

- Plug in RAM cartridge gives you up to 16K EXTRA MEMORY!!
- That's over 28K free for Basic and even an extra 18K in HiRes mode!
- Plugs in to memory expansion port - then just switch on.

ONLY £14.99

256K SUPEROM EXPANDER

- Select instantly from 8 sockets which accept up to 32K EPROM each.
- Program your own EPROMs using our EPROM programer.
- No need to have lots of cartridges - just make a selection from the Superom menu.
- Fully menu driven on power up.
- Select any slot under software controls.
- Unique EPROM generator feature will take your own programs - basic or m/c & turn them into autostart EPROMs. (EPROM burner required).
- Accepts 2764/27128/27256 EPROMs.
- On board unstoppable reset.
- On board operating systems - no programs to load.

ONLY £29.99

ALL ORDERS NORMALLY DESPATCHED WITHIN 48 HRS

HOW TO ORDER ...

BY PHONE	BY POST	FAX
0782 744707 24hr Credit Card Line	Send cheques/POs made payable to "Datel Electronics"	0782 744292 UK ORDERS POST FREE EUROPE ADD £1 OVERSEAS ADD £3

PRICES AND SPECIFICATIONS CORRECT AT TIME OF PRESS AND SUBJECT TO CHANGE WITHOUT NOTICE

CALLERS WELCOME - Please reserve goods by telephone prior to visit.

DATEL ELECTRONICS

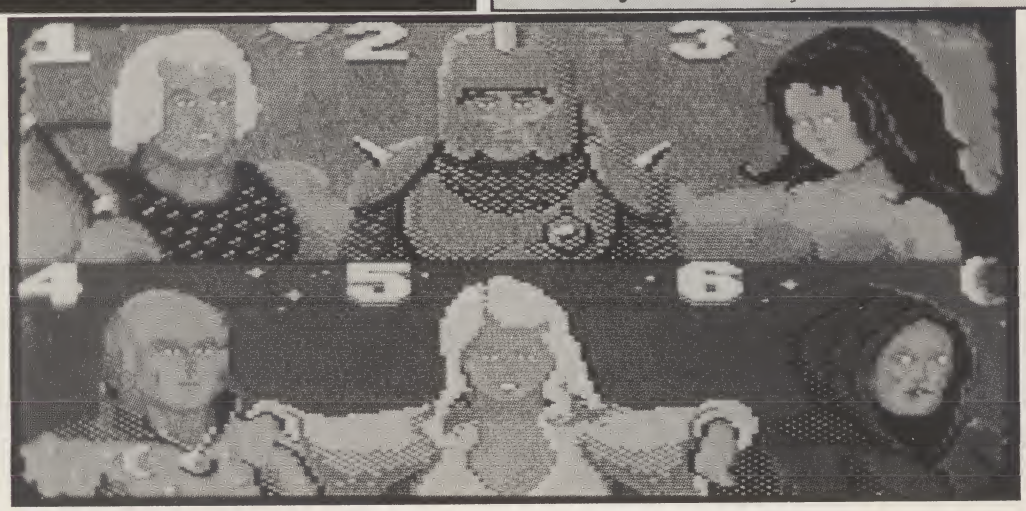
DATEL ELECTRONICS LTD., FENTON INDUSTRIAL ESTATE
GOVAN ROAD, FENTON, STOKE-ON-TRENT, ENGLAND.

SALES ONLY	TECHNICAL ONLY
0782 744707	0782 744324

IN THE MAGAZINE

ON THE DISK

Group Editors: Stuart Cooke/Mark Webb
Editor: Paul Eves
Technical Editor: Paul Eves
Production Editor: Alison Smith
Cartoonist: Alan Batchelor
Photography: Manny Cefai
Adventure Correspondant: Gordon Hamlett
Advertisement Manager: Paul Kavanagh
Display Sales Exec: Maria Wade
Classified Sales Exec: Jayne Heigold
Designer: Mark Newton
Origination: Ebony Typesetting
Distribution: S. M. Distribution
Printed by: Chase Web, Plymouth



Editors note

We have a mixed bag of programs for you this issue. For the collectors, we have provided a nice SCRAPBOOK program. For the machine code freaks there is a very good MONITOR. On the graphic side, we examine those HIDDEN GRAPHICS capabilities of the 64. Those of you that like to keep neat records and files may find SUPER FILE of some use. On the other hand, if you are a rather untidy disk keeper, then DISK HUNTER is just up your street. On the entertainment side you will find four very good games. COBALL, CELLRATOR, FORTRESS and RAINBOW CHASER. You may notice a distinct lack of a C128 program in this issue. We can only publish what we receive, so all you 128 users out there, get writing!

How to copy CDU files

You are welcome to make as many of your own copies of Commodore Disk User programs as you want, as long as you do not pass them on to other people, or worse, even sell them for profit.

For people who want to make legitimate copies, we have provided a simple machine-code file copier. To use it, simply select the item FILE COPIER from the main menu. The copier works with a single drive, is controlled by means of the function keys as follows:

F1: Copy file – the program will prompt you for a filename.

F3: Resave the memory buffer – you may get an error on a save (perhaps you left the drive door open). Use this to try again.

F5: Disk commands – allows you to enter any regular C64 disk command.

F7: Displays the directory.

F2: Exits the program and returns you to basic.

Disk Instructions

We do our best to make sure that Commodore Disk User will be compatible with all versions of the C64 and C128 computers.

Getting the programs up and running should not present you with any difficulties, simply put your disk in the drive and enter the command.

LOAD "MENU",8,1

Once the disk menu has loaded you will be able to start any of the programs simply by pressing the letter that is to the left of the desired program.

C128 users please note that you should be in C64 mode when using the disk. You can enter this mode by either:

1) Holding down the Commodore key (bottom left of the keyboard) when turning your computer on or,

2) After turning the computer on type G064 and anser 'Y' when prompted "ARE YOU SURE".

It is possible for some programs to alter the computer's memory so that you will not be able to LOAD programs from the menu correctly until you reset the machine. We therefore suggest that you turn your computer off and then on before loading each program.

Disk Failure

If for any reason the disk with your copy of Disk User will not work on your system then please carefully re-read the operating instructions in the magazine. If you still experience problems then:

1) If you are a subscriber, return it to:
INFONET LTD
5, River Park Estate
Berkhamsted
Herts. HP4 1HL
Telephone: 0442 876661

2) If you bought it from a newsagents, return it to:
CDU Replacements
Direct Disk Supplies
Unit 19
Teddington Business Park
Station Road
Teddington
Middlesex: TW11 9BQ
Telephone: 01 977-8777

Within eight weeks of publication date disks are replaced free.

After eight weeks a replacement disk can be supplied from DDS for a service charge of £1.00. Return the faulty disk with a cheque or postal order made out to DDS for £1.00 and clearly state the issue of CDU that you require. No documentation will be provided.

Please use appropriate packaging, cardboard stiffener at least, when returning disk. Do not sent back yor magazine – only the disk please.

Back Issues

Back issues of Commodore Disk User are available at £3.00 per issue, via:

Infonet Ltd
5, River Park Estate
Berkhamsted
Herts
HP4 1HL
Telephone: 0442 876661

Those magazines available are:

November/December 1988: Utilities – CDU Forth, Texted, Extractor, Windows 64, ZMON 128. Games – Oblivion, Cribbage Master.

January/February 1989: Utilities – Easy Scroller, Data Maker, Border Sprite, Disk Turbo, Menu Maker 128. Games – Blastball, Microdot, Runaway, Colour Bind, Logic, Spots, Life.

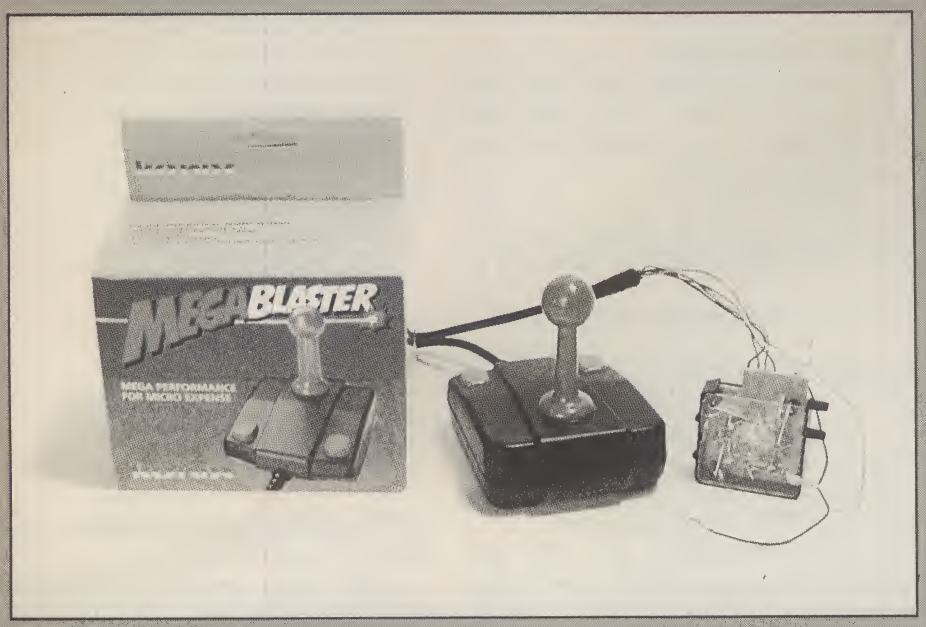
March/April 1989: Utilities – CDU Paint, Devaid, 128 Graphics Primer. Games – Darts, Bazair, Araknifoe, Domines, Phantom.

May/June 1989: Utilities – Sid Sequencer, 6510+ Assembler, Sound FX Kit, Base-ed, High Speed Graphics, Dbase 128, Games – Liberte.

July/Aug 1989: Utilities – 64 News Desk, Animator, Border Msg Scroll, Font Factory 89, Hires Demo Kit, Screen Copy Master, Texted(Update), Vidibasic, Typit-128.

Update

The latest gossip from the disk world

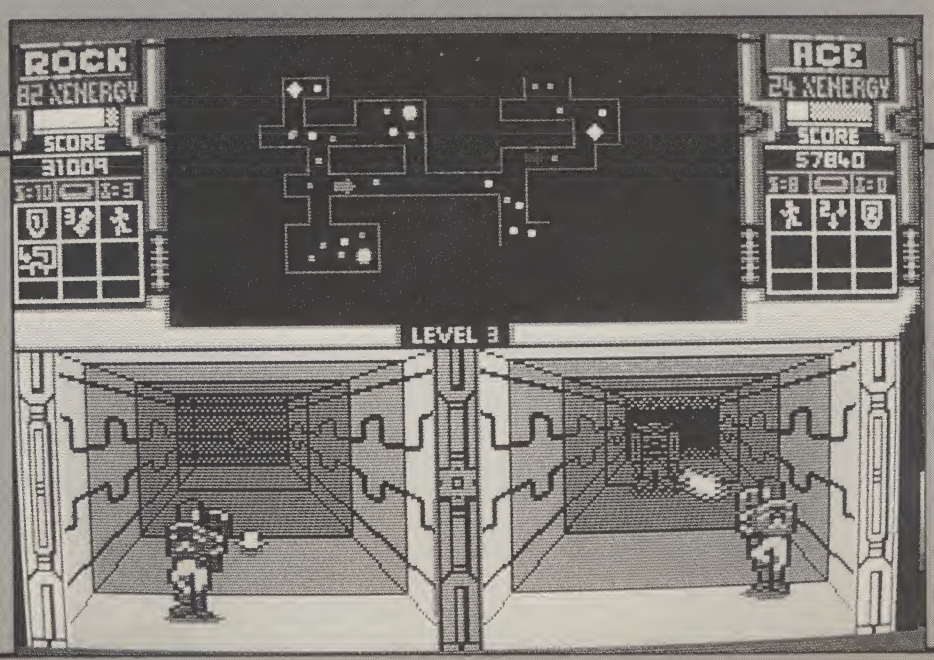


Switch your joystick

Konix, the joystick specialists, have released their popular Megablaster with microswitches. Not only can you now have a very strong and resilient stick in your hands, but you can experience a far greater accuracy. The new megablaster retails for £8.99 and to quote a Konix Director, "Never before have games players had the opportunity to obtain such a high quality joystick at a price as low as this".

The time is now

The latest coin-op conversion from **Domark** is *Xybots*. The game features a split screen for two players, which can give their own independent viewpoint. The game consists of going around the maze and collecting wealth. With this wealth you purchase your much needed bonuses, like speed, power of shots and a map. (Without the map you have no chance). Of course, preventing you from achieving your goal are the metallic monsters themselves.



Green Joysticks??

Microprose and Greenpeace have joined forces to bring us 'Rainbow Warrior'. Is this the first ever environmentally friendly computer game? No nasty aliens or space ships to blow to Kingdom come here. No sir! instead, you have to protect wildlife and stop the pollution of the natural world. I suppose in view of all the recent hysteria, this is a timely release. However, whether or not it transports over successfully to the computer screen remains to be seen. We are informed that the graphics are not only outstanding, but are also Python-esque, with large sprites and smooth animation. Prices will be £24.95 for 16-bit machines and £14.95 (disk) and £9.95 (cassette) for 8-bit formats.

Politics Vs Children

That popular and rather nicely presented educational program – *Fun School 2* – has come under attack from certain narrow sighted politicians. it is becoming apparant that certain authorities are trying to outlaw the use of this program as a teaching aid. Far be it for me to get involved in political arguments, or opinions, but I do feel that these 'lunatics' are taking things a bit too far. After all, as the title suggests, this is a FUN program. (Ask my 6 year old son Christopher, he loves it – both the C64 and the Amiga version). Come on you so called 'trendies', leave the kids alone and let them have some fun.

Mars trip postponed

Arcana Software Ltd have regrettably had to postpone their intended release date for *'Mars Cops'*, due for release in late June, until sometime in September. The game has hit upon some unexpected technical problems. We will keep you informed of it's progress.

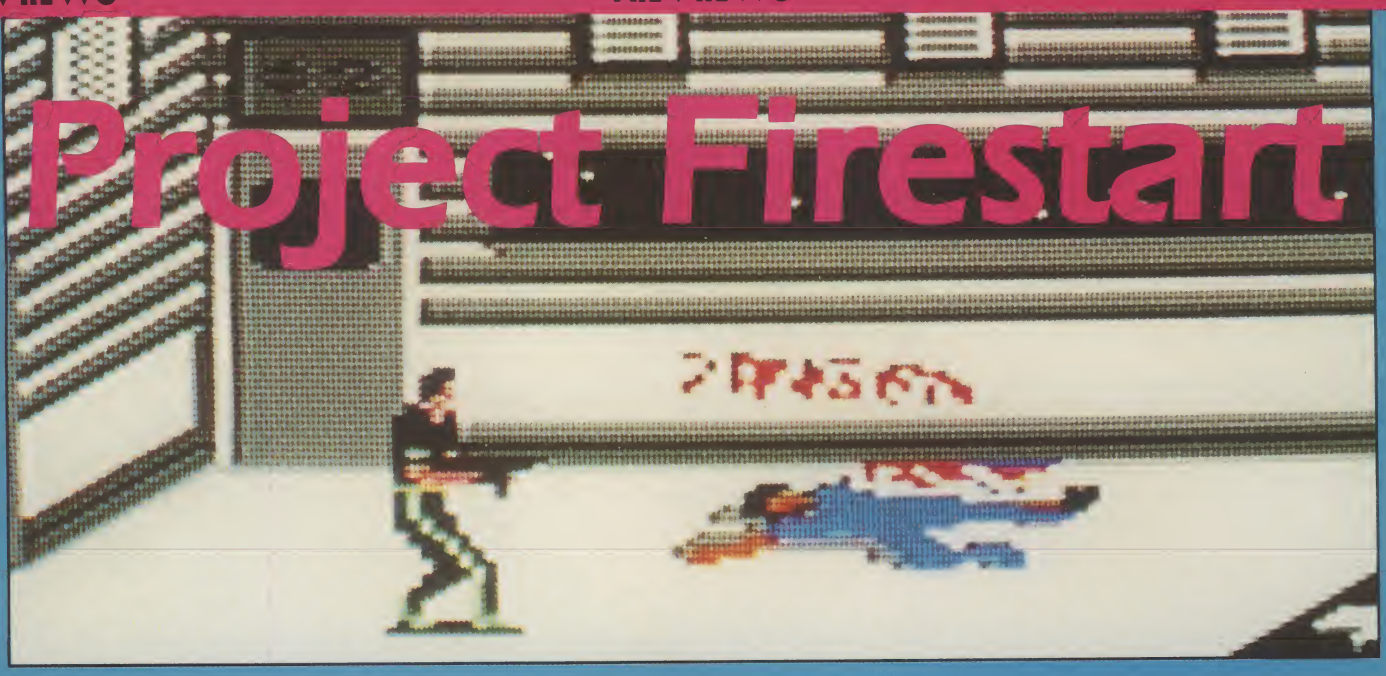
One for your Diary

Database Exhibitions, those Macclesfield based exhibition organisers are staging the Computer Shopper Show, to be held at The Great Hall, Alexandra Palace. This will take place from 24-26 November. The show will be host for displays and demonstrations of all the latest hardware, software and peripherals for all the popular make of home and business computers. There are an estimated 200 exhibitors attending. Could be well worth a visit.

Get it right

The Activision Hot Line, which provides consumers with hints and tips on all of their games, plus information on forthcoming releases, has been misprinted in some magazine titles. This misprint is causing a poor Reading resident some concern. If you would like to contact this hot line please note the correct number: Reading (0734) 310003

Project Firestart



Project Firestart began in the research labs of the SSF (System Science Foundation) ship Prometheus. The plan was to genetically engineer strong durable labourers that would be capable of helping Belters mine titanium and iridium on the more hostile moons and asteroids. This is, of course, incredibly dangerous but stringent safety measures were introduced. However, these have obviously been breached as contact was lost with Prometheus which now must be out of control. You're being sent in by the SFF to sort ouf the mess.

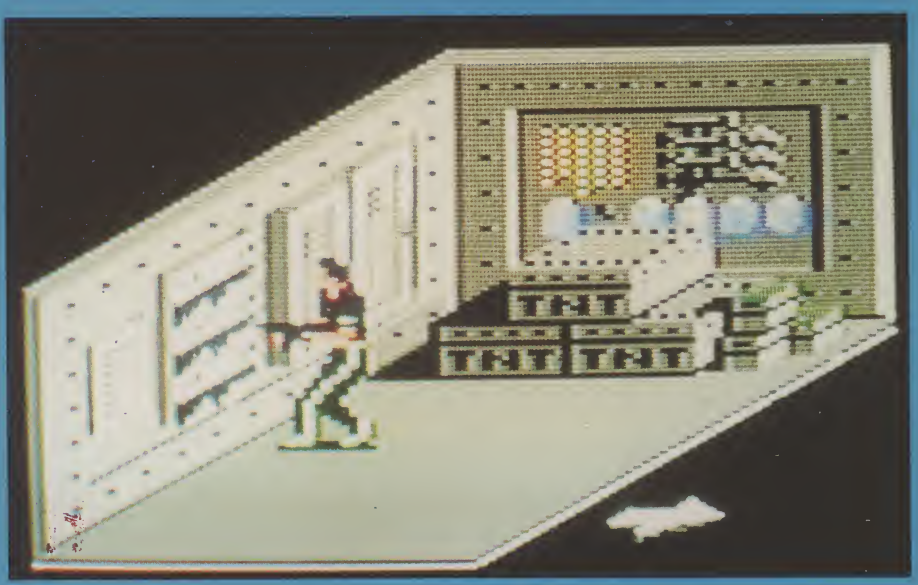
As an SFF terminator you're used to wielding some impressive firepower but this time that has to be left on your ship for fear of blowing a hole straight through the hull so you're restricted to rather puny lasers, one of which you carry, the others you'll have to find on the ship.

The ship is displayed on screen as a series of isometric 3D rooms and corridors that are handily labelled to tie in with the maps included with the instruction card. They contain the usual collection of labs, cargo, life support and engineering and science areas that you'd expect on a research ship but it also contains numerous extremely bloody and gory bodies of the crew and very large, strong and unpleasant aliens that will leap out and mangle you at every opportunity.

Although the game looks like a shoot-em-up it actually plays like an arcade adventure and it's important to remember to save the game frequently as you only have one life. Having said that the game is almost entirely controlled by joystick with only four keyboard commands to change weapons, access the disk (load and save), Inventory and pause. Whenever, you approaching a door, object or interesting corpse separate screens or options will appear that can be selected by joystick.

As the game unfolds you have to be continually on your guard for alien attack while searching for security passes, new lasers and other useful items that might let you survive long enough to save the cryogenic survivors from becoming frozen mutant food.

The result is a very tense game packed with atmosphere that is undoubtedly helped by the sinister soundtrack that plays in the background.



At a glance

Title: Project Firestart.
Supplier: Electronic Arts. 11/49 Station Rd, Langley, Berks., SL3 8YN. Tel: 0753 49442
Price: £14.95
Graphics: Isometric 3D rooms full of objects, aliens and corpses.
Sound: The sinister soundtrack piles on the atmosphere.
Playability: Easy to play but also very easy to die.
Addictiveness: As you survive for longer and get further in the game you'll want to waste more and more mutants.

Stormlord

Stormlord the hero is the last hope for the land. It was a wonderful land full of fairies but now they lie helpless in giant cages at the mercy of an evil queen. It's up to you to find a way to free them and ultimately the means of destroying the malevolent monarch before she pummels the land into eternal terror.

The game is played over a number of increasingly difficult levels constructed from a number of sideways scrolling screens. Each screen is itself compiled from platforms, walls, statues and steps that our hero must negotiate by standard joystick movement for walking and crouching a jump that increases in power the longer it is held in place. So with practise you should be able to maintain absolute control of where you will land. That's the theory anyway and it's complicated by minions of the evil queen as well as the natural hazards that are turning it into a dangerous place to be.

Venus fly traps lurk to chomp the unwary but more aggressive foes including giant worms, dragons and hideous insects would like to add you to their menu and so they must be destroyed with your self loading flying sword. One way to get across a lot of land in a hurry is via the network of springboards. Simply jump on the end of one and you'll be thrown through the air until you land safely several screens from your launch position. These springboards have set routes so you can't choose where you want to go but it's also a safe bet that you'll find one for the return trip.

That game is a return to the old style of arcade adventuring where you have to find objects (and you can only carry one at a time) and also discover when and where to use them. Some are obvious such as a giant key which just might gain you access through a door but what of a vase and an umbrella.

It's important to remember that your objective is to release the imprisoned fairies which is achieved by touching them but some are harder to get at then others where a loss of at least one of your six lives looks inevitable.

At the bottom of the screen a display shows your lives remaining, the time you have, (illustrated by a sun changing into the moon) the object you are currently carrying and the number of fairies you have liberated and those left to be freed to gain access to the next level.

If you're looking for simple but addictive gameplay then this is the game for you.

At a glance

Title: Stormlord
Supplier: Hewson, Hewson House, 56B Milton Park, Milton, Oxon, OX14 4RX. Tel: 0235 832939
Price: £14.99
Graphics: Sideways scrolling platforms and perils.
Sound: Stirring soundtrack to slay by.
Playability: Easy to learn but difficult to master.
Addictiveness: I'll just have one more game...

Disk Dungeons

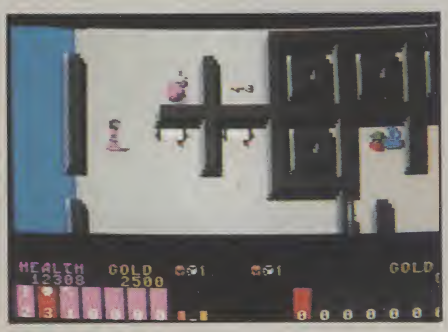
Not a great deal to report about in what is traditionally a quiet time of the month. The only releases that I have seen are two role playing games, one very good but perhaps looking a trifle jaded now, the other pretty awful.

I am now approaching the end of my complete solution to *Dungeons and Dragons*. As I have mentioned before, I am hoping to give some help with *Ultima V* in a future issue and would be more than grateful for any tips from our readers. The only response so far has been from a little old lady, Doreen Sellman, so if she can be bothered to write in, why can't you? Send all your hints to me, Gordon Hamlett, c/o Commodore Disk User. There will be a prize for the best entry.

FIRE KING

The story is a familiar one. A sleepy community is ruled by four Mages, each one representing one of the elements: air, earth, fire and water. Life isn't perfect but on the whole, it's pretty reasonable and to date, no-one has seen fit to complain.

No-one, that is, until the last conclave of the wizards. As usual, there were weeks of petty bickering beforehand as minor magicians tried to score political brownie points. Again though, this was all fairly normal. The news of the Fire King's gruesome death stunned the town, especially when coupled with the news that the Water Mage had disappeared and the Earth Mage had fled to her forest retreat, apparently insane.



It was a grievously wounded Wind Mage that brought the news of an attack by some super-human magical beast. This beast was killed in the battle with the magicians but another one soon appeared to take residence in the town's catacombs, appearing occasionally to feast on one of the townsfolk.

The mayor of the town persuaded one citizen to see if he could fight off the beast once and for all. He never returned. Now it is your turn.

The game starts with you deciding which one of six characters you wish to play. Each one has different armour, strength and magical abilities, together with some gold and a loaded crossbow. Initial values for your three attributes start off fairly low and you will need to find many ancient relics on your travels – the only way of permanently boosing their level.

Fire King is a curious hybrid. Although it possesses many of the features of a role playing game such as a quest, puzzles to be solved and spells, it looks and plays more like an arcade game and it would not be too far from the truth to state that *Gauntlet* immediately springs to mind. As you wander round, there are many monsters all appearing from generators. I wonder where we have seen that before?

There are many objects to be picked up and manipulating them is the key to the whole game. You only have seven pockets in which to store things, although each of these can hold up to nine similar items eg keys, death spells etc. The problem is that there are more than seven different types of item and so you must decide what to take and what to leave behind. I found using the objects in the 'real time' combat situations to be clumsy. By the time I had selected the correct pocket and used a spell or whatever, the monsters had drained half my strength.

There were far too many monsters as well. It became a pain trying to shop and fend off assorted beasties at the same time. This I felt summed up the whole game. The balance was all wrong. Role playing fans don't want all the action, they enjoy sitting around planning what they are going to do next and developing different strategies when it comes to battles. They do not want a shoot-em-up. Similarly, there is likely to be too much thinking involved for arcade fans. Couple all this with a clumsy gameplay and I was left very disappointed, especially considering that the parent company, **SSG**, have produced some excellent and highly rated wargames in the past. I was hoping for something a little better than this mish-mash.

Title: Fire King
Supplier: Electronic Arts
Price: £18.95

ULTIMA TRILOGY

Regular readers of this column would have to be pretty dense if they failed to realise that I am a great fan of the *Ultima* series of role playing games from **Origin.** The first game to be released in this country, some years ago, was Ultima III, followed by IV and V. Ultima I was released briefly and, as far as I know, Ultima II has never been released. To remedy matters, Origin have now released Ultima I, II and III as a trilogy – surely one of the bargains of the year.

To be fair, these games are beginnig to show their age somewhat but nevertheless, they can still show their rivals a thing or two. What is also fascinating, is to watch the development of the series. Each new game is approximately twice the size and has twice the detail of its predecessor.

One of the best parts of the series is that the stories have a cohesion that is often lacking in rival products. The

game does not just consist of a hack and slash dungeon but takes in a whole world complete with different countries, towns, castles as well as a whole array of underground scenarios. It must also be mentioned that the first two games in the series feature space travel and combat as well!

Ultima I is subtitled the First Age of Darkness. The evil Wizard Mondain is causing havoc all over the fair land of Sosaria. You have been summoned to rid the world of this menace but it will not be easy. The source of Mondain's power is his staff and you will need to travel back in time in order to confront him. There are problems too in the present. The only safe places appear to be the castles but the rulers often compel you to accomplish other tasks first.

In the second age of darkness, you have to overcome Mondain's assistant, Minax, an enchantress with telekinetic powers. Again, you will have to travel through five different time zones ranging from the age of legends to a post holocaust period as you try to destroy this evil being once and for all.

Your task in the third game is to destroy something called Exodus – a tricky task since nobody knows who or what it is. All that is known is that an island, complete with castle, is rumoured to have appeared in the middle of the ocean.

Control of all three games is roughly the same. Commands are entered via a series of key strokes. These are all fairly easy to remember and there is a high degree of continuity between the games. Ultima I and II feature a single character – you can choose race and character class etc as well as tinker with your player's characteristics. In Ultima III, you control a party of up to four characters. These can either be created at the start of the game or you can try and recruit three others to your cause as you progress.

Magic features in all the games and again, develops considerably throughout the series so that in Ultima III, there are some thirty two different spells spread across wizard and cleric classes. The spells are all unique too. So many games have variants on the small, medium and large damage/heal spell ie only a few spells but with varying degrees of potency.

One final feature of the games is the accompanying documentation. There is a 100 page book detailing the storylines, monsters and general background information. Two eight page reference guides and quickstart instructions list with all the commands that you will need. Finally, three maps detail some but not all of the area to be explored. The only thing missing is a translation scheme for the runes but a bit of native intelligence should help here. All in all, you can't help but feel that you are getting value for money.

This package really does represent superb value for money and if you don't already own the games, I would suggest that you go straight out and buy them. While you're at it, buy Ultima IV and V as well. They all knock spots of their rivals.

Title: Ultima Trilogy
Supplier: Origin, Microprose
Price: £24.95

Dungeons and Dragons – Pool of Radiance The Solution continues

There are three more quests to be completed in the wilderness. The Buccaneer's Base involves the rescue of a kidnapped heir. There are two ways to complete this scenario. The first is to fight your way in and out. The more subtle method is to find where the boy is being held, stampede the animals, grab the boy and escape in the ensuing chaos.

The outpost of Zhentil Keep is another fairly simple episode involving a considerable amount of diplomatic double dealing. Learn as much as possible at the evening meal and make sure that you post a guard before going to sleep at night. Then it is simply a case of making good your escape and reporting what has happened on your return to the council. The longer you stay, the more battles you will have to fight.

Your quest on the Sorcerer's Island is to put a stop to the pollution that is contaminating the river. It is the part of the game that gave me the greatest problems, not least because it features that bane of all dungeon mappers, the teleport. In fact, there are three different teleport systems and as if that wasn't bad enough, one of them is two way i.e will teleport you to one of two different locations depending on whether the 'switch' is set to forward or backward. It is easy to tell the two way systems as they are the ones with the pile of stones beside them. Throw a stone to change the direction setting.

There are three levels to the pyramid. At the bottom is a central passage complete with mazes on either side. In the maze you might fine a partially completed map or a mad priest who may show you the way out if you talk nicely to him. There are also many half starved monsters intent on eating you.

On the second level, there are many mutant lizardmen and not a lot else. On the third level, You must use the password 'NOTNOW' or your party will be destroyed. When you come to the machinery room, break it once and then run or you will get caught in an explosion. Talk nicely to the lizardmen here to learn a password that might help you with other lizardmen in different scenarios. In Yarash's office, after you have defeated him, search the room carefully. The four settings on the teleport take you to three treasure rooms and the exit.

Back on the mainland, there is the graveyard to be tackled. This is a sort of no-go area as the Boss is scared of sending his own men into this area in case they should be turned. The Undead are here in huge numbers but you should have a magic sword given to you by the council, together with several restoration spells to be used if one of your characters loses an experience level in combat with a wraith or similar. Always remember to have your clerics try to turn undead in all combat situations.

All the minor undead are being created by spectres so it is a good idea to get rid of these as soon as possible. Cast spells that raise your saving throws before fighting either them or the mummies. There is a lot of treasure around so walk round in search mode – you are going to have to fight the monsters anyway. Don't let the magician join your party. He is evil and will join the vampire to attack you. The vampire has to be killed twice in order to destroy him completely. Use the efreeti bottle from the kobold's cave to help you.

You are now ready for an assault on the castle itself, once you have stormed the Stojanow Gate. This is simply a case of beating up all the guards. You can buy a disguise to help you sneak up but I didn't really find it necessary. You will however need two 'knock' spells to get you through the gates.

Next month, the final assault.

Correction For Introducing Sid

Once again the dreaded gremlins have been at work. This time it concerns the article in the May/June issue of CDU, titled *'Introducing Sid'.*

The error in question is quite a drastic one, unless you happen to own a US Version of the Commodore 64. We published a table of notes and their respective values. Unfortunately, these notes are for the American versions of the 64. The reason for this is that the UK version runs at a different clock speed. Thus, if you use the table as provided, your sounds will be at least one semitone out.

The 64 derives the note frequency from the system clock. In the U.S., the clock runs at 1.02273 MegaHertz (MHz). In the UK., the clock speed is 0.98525 MHz (see page 351 of the '64 Programmers Reference Guide).

Therefore, to calculate the value to give SID, use the formula:-

$$\text{SID Value} = \frac{\text{Frequency}}{\text{Magic Number}}$$

The magic number is derived as follows:

$$\text{Magic Number} = \frac{\text{Clock Frequency}}{(16*1024*1024)}$$

The U.K. magic number is 0.058725476
The U.S. magic number is 0.060959458

We apologise for this error and hope that it does not spoil any work that you may be creating at the moment.

I would like to thank **Mr. Brian S. Craigie.** BSc, of Livingston, Scotland, for pointing out the error to me in the first instance...

Octave	Note	Freq.	Sid	Hi	Lo	Octave	Note	Freq.	Sid	Hi	Lo
0	c	16.4	278	1	22	4	c	261.6	4455	17	103
0	c#	17.3	295	1	39	4	c#	277.2	4720	18	112
0	d	18.4	313	1	57	4	d	293.7	5001	19	137
0	d#	19.4	331	1	75	4	d#	311.1	5298	20	178
0	e	20.6	351	1	95	4	e	329.6	5613	21	237
0	f	21.8	372	1	116	4	f	349.2	5947	23	59
0	f#	23.1	394	1	138	4	f#	370.0	6301	24	157
0	g	24.5	417	1	161	4	g	392.0	6675	26	19
0	g#	26.0	442	1	186	4	g#	415.3	7072	27	160
0	a	27.5	468	1	212	4	a	440.0	7492	29	68
0	a#	29.1	496	1	240	4	a#	466.2	7938	31	2
0	b	30.9	526	2	14	4	b	493.9	8410	32	218
1	c	32.7	557	2	45	5	c	523.3	8910	34	206
1	c#	34.6	590	2	78	5	c#	554.4	9440	36	224
1	d	36.7	625	2	113	5	d	587.3	10001	39	17
1	d#	38.9	662	2	150	5	d#	622.3	10596	41	100
1	e	41.2	702	2	190	5	e	659.3	11226	43	218
1	f	43.7	743	2	231	5	f	698.5	11893	46	117
1	f#	46.3	788	3	20	5	f#	740.0	12601	49	57
1	g	49.0	834	3	66	5	g	784.0	13350	52	38
1	g#	51.9	884	3	116	5	g#	830.6	14144	55	64
1	a	55.0	937	3	169	5	a	880.0	14985	58	137
1	a#	58.3	992	3	224	5	a#	932.3	15876	62	4
1	b	61.7	1051	4	27	5	b	987.8	16820	65	180
2	c	65.4	1114	4	90	6	c	1046.5	17820	69	156
2	c#	69.3	1180	4	156	6	c#	1108.8	18880	73	192
2	d	73.4	1250	4	226	6	d	1174.7	20002	78	34
2	d#	77.8	1324	5	44	6	d#	1244.5	21192	82	200
2	e	82.4	1403	5	123	6	e	1318.5	22452	87	180
2	f	87.3	1487	5	207	6	f	1396.9	23787	92	235
2	f#	92.5	1575	6	39	6	f#	1480.0	25202	98	114
2	g	98.0	1669	6	133	6	g	1568.0	26701	104	77
2	g#	103.8	1768	6	232	6	g#	1661.2	28288	110	128
2	a	110.0	1873	7	81	6	a	1760.0	29970	117	18
2	a#	116.5	1984	7	192	6	a#	1864.7	31752	124	8
2	b	123.5	2103	8	55	6	b	1975.6	33640	131	104
3	c	130.8	2228	8	180	7	c	2093.0	35640	139	56
3	c#	138.6	2360	9	56	7	c#	2217.5	37760	147	128
3	d	146.8	2500	9	196	7	d	2349.3	40005	156	69
3	d#	155.6	2649	10	89	7	d#	2489.0	42384	165	144
3	e	164.8	2806	10	246	7	e	2637.0	44904	175	104
3	f	174.6	2973	11	157	7	f	2793.8	47574	185	214
3	f#	185.0	3150	12	78	7	f#	2960.0	50404	196	228
3	g	196.0	3338	13	10	7	g	3136.0	53401	208	153
3	g#	207.6	3536	13	208	7	g#	3322.4	56575	220	255
3	a	220.0	3746	14	162	7	a	3520.0	59940	234	36
3	a#	233.1	3969	15	129	7	a#	3729.3	63504	248	16
3	b	246.9	4205	16	109	7	b	3951.1	unobtainable		

Battletech

You've no doubt played one of the many film and TV licences but now you've played the game of the film and the game of the series, now try the game of the game. **Battletech** is based on the futuristic FASA game in which giant mech fighting machines clash in combat. It also marks a dramatic change in direction by *Infocom* which was up to now, only interested in text adventures.

The benefits of using a boardgame as a base for a game are obvious as its game system has already been tried and tested and adding computer action and graphics can only improve it. In this the boardgame provided the background and combat system, *Infocom* added the plot.

You take the role of young Jason Youngblood, trainee mech warrior and son of the legendary Jerimiah Youngblood. Your living is comfortable and free as you're the good guys known as the Lyran Commonwealth. The bad guys are the Kuritans and they're about to stomp all over you so it's about time you got some training done. This, let's face it, rather tedious process involves saving up your credits to afford weapon skill classes and a series of essential mech training sessons. These range from learning simple controls to combat with one, two and three other mechs. Which is quite ironic as just as you're beginning to understand the tactics, the Kuritans attack and you're thrown out into the cruel world, on foot.

Your first moves are to reach the Starport as you reckon this is the best place to meet other rebels, which is what happens, and luckily one of them owns a mech so you can start blasting enemy mechs with your machine guns, lasers and missiles while keeping a careful eye on your heat sinks as if they overheat your mech will shutdown and become a giant sitting duck.

However, the game is more than just a combat game as your aim is more ambitious as you must organise the remaining rebel force (this involves springing one from jail) and battle your way through to a secret horde of mech parts that you can use to create a super army and win the day. Before then you'll have to fight countless battles, learn how to salvage mech parts and ammo, discover the Kuritans masquerading as rebels before they damage your cause and complete a series of mini adventure sequences that will lead to the location of the cache of mech parts.

Battletech is a big game with a claimed total number of locations of over 4 million containing mechs and humans to fight and towns to explore.

At times the combat system is quite slow as you must individually target every one of every mech's weapons and plot each step of movement but it does give you a chance to devise your own strategy. The rest of the game moves at quite a pace and you'll find yourself spending hours at the keyboard. The only problem is the tedious training session which is mind numbingly dull. However, don't despair as the game dramatically improves once the Kuritans attack.

At a glance

Title: Battletech.
Supplier: Infocom (Activision), Blake House, Manor Farm Road, Reading, BERKS., RG2 0JN. Tel: 0734 311666
Price: £19.95
Graphics: 4 million locations of "sort of" top down views.
Sound: Bleeps and bangs.
Playability: Response is quite slow but you'll get used to that.
Addictiveness: If you survive the tedium of the training session you'll be hooked.

Graphic Interrupt Routines

An insight into achieving those scrolling routines and raster bars

By Miles Barry

This article covers various routines that can be accessed using the *VIC-II CHIP*, and the colour RAM. The routines are outlined below:-

1. The use of the raster line and what effects can be done with it.
2. The use of the colour RAM to produce colour scrolling.
3. The use of the smooth scroll registers. ($DO11 & $DO16)
4. General information on the timing of the raster scan.
5. Multi-coloured bars, using the raster, and how to produce movement effects using it.

Firstly, the Raster Scan. A Raster Scan is an electron beam that sweeps through the screen display, lighting up tiny areas of the screen known as pixels. If the pixel is set to 1 then it is on, if it is set to 0 then it is off. The Raster Scan starts from the top left hand corner of the screen and travels down and over to the bottom right hand corner. There are 312 raster lines to a screen, and any of these lines can be latched onto.

The Commodore 64 has a raster compare register at $D011 & $D012. (The raster compare register is really 16 bits long, as the 9th bit overlaps into $D011). Most people only use $D012 to check for a raster line, as this allows you to latch onto lines 0 to 255.

You may, or may not know, that each machine language instruction takes a certain amount of time for the CPU to process it. If you were to write a piece of code to latch onto a raster line, there would be the 'Well Timed' way and the not so 'Well Timed' way. The well timed code would look something like the following:-

```
LDX # 41      :- The raster line
AG CPX $DO12  :- Check for x in comp.reg
BNE AG        :- If not line, go back.
```

The poorly timed code would look something like the following:-

```
AG LDA $D012  :- Load raster line into A
CMP # 41      :- Is it 41?
BNE AG        :- If not redo from start
```

In the first example, we have taken less time to locate the raster line. Whereas in the second example, we keep loading the accumulator with the raster compare register. This wastes valuable time. (Raster line 41, is the top of the screen, excluding Border, just in case you were wondering!)

Every eighth raster line, a DMA line is generated, this prepares the screen for the next 8 lines. To a DMA line you have 23 cycles (Each instruction takes a certain amount of cycles/time), but to a normal raster line, there are 63 cycles. When you latch onto a raster line, it slows the speed of the raster scan, this causes screen flicker, so to avoid this it is necessary to write code that takes as few cycles as possible.

I don't think it is necessary to go into depth about the timing and such like, but if you do wish to learn more regarding the raster scan, and/or learn machine language, then a book by **Rae West,** called ***Programming the Commodore 64,* published by Level Computer Publications,** priced at £14.90, is well worth looking at. The ISBN number for the book is:- 0-9507650-2-3.

Another book that's useful is *Commodore 64 Sound & Graphics,* written by **Peter Falconer,** ISBN number is: 0-86161-144-6.

All the above information is important, but if you feel that you are having problems understanding any of it, then do purchase some book (Preferably the one by **Rae West**), as this goes into more depth regarding raster scan, and other items.

It is now time to start explaining colour scrolling, as this isn't hard to understand, and is easy to code.

Colour scrolling is the art of rotating, let's say for example, the top line of colours, in a loop.

The theory for scrolling left, the top of the colour display is as follows:-

A. You take the left hand byte of colour (Nearest the border), and replace it at the right hand end (Nearest the border on the right).
B. Now you shift the coloured line one place to the left.
C. You repeat the process.

This theory will scroll any colours, that are on the top line round in a loop. The code for this would look like the following:.

```
L0 LDA $D800  :- Take left hand colour
STA $D827     :- Store at end of line
L1 LDX # $00  :- Set X register to 0
```

C64 PROGRAMMING

```
LDA $D801,X   :- Load A with colour
STA $D8000,X  :- Shift colour one left
INX           :- Increase X register
```

```
CPX # $27     :- Is X $27?
BNE L1        :- If not, goto L1
JMP L0        :- Re-run routine
```

If you assemble this routine into an assembler (You could use 6510+), and put colours on the top line, then SYS the routine, all the colours will scroll left in a loop. If you thought all that information was too complex, then stop reading, because it gets even more tricky now.

Have you ever seen a game or demo, where a logo, or text has a wobbly effect running through it? If so, you probably have wondered how you could do such a routine yourself.

The wobble effect is achieved by using the horizontal smooth scroll register at $D016. This allows you to scroll the screen display either 8 pixels left or 8 pixels right.

If we were to latch onto raster line 41 (Decimal), and then constantly change the smooth scroll value, we would see the text shift into say a 45 degree effect. If we then rotated the wave up constantly, it would give the impression that there is a wave going through the text. That information might seem a bit long winded, so I will give you a picture, that explains it.

```
0000000*
000000*0
00000*00
0000*000
000*0000
00*00000
0*000000
*0000000
0*000000
00*00000
000*0000
0000*000
00000*00
000000*0
0000000*
```

The star represents the 45 degree effect, so if we were then to rotate this up one, it would look like this.

```
000000*0
00000*00
0000*000
000*0000
00*00000
0*000000
*0000000
0*000000
00*00000
000*0000
0000*000
00000*00
000000*0
0000000*
000000*0
```

So if we keep rotating the values up one at a time, we get the wave effect.

Now I will explain how raster bars work, and how you can use them as part of your programs.

Once again, we are dealing with the raster scan, and the raster compares register. To achieve our aims, the necessary routine is basically a modified form of the wave routine, but instead of changing the horizontal smooth scroll register, we change the border and paper colour.

As the raster scan is so fast, it is possible to change the border and paper colour in time with the raster, to produce a coloured set of bars the full width of the television/monitor screen. I will explain the step by step method of producing still raster bars, as opposed to animated ones.

A. Firstly we must latch onto the desired raster line using $D012 raster compare register.
B. We must load the desired colour into the accumulator.
C. We must store the colour, which is held in the A register, at $D020 (53280 Dec) and $D012 (53281 Dec). **Note**:- These are the border and paper colours.
D. We must re-run the routine, until all the desired amount of colours are placed onto the screen.
E. When all bars are on the screen, we must set the paper and border colour to black, or the colour you want the screen to be.

Here is a short routine that you could try writing in 6510+. The routine will place a white bar through the screen display.

```
SEI           :- Disable interupts.
LP0 LDA # $01 :- Set colour white.
    LDX # $29 :- Set Raster line.
LP1 CPX $D012 :- Check raster line.
    BNE LP1   :- Is it correct??
    STA $D020 :- Set border colour.
    STA $D021 :- Set paper colour.
    LDA # $00 :- Set colour black.
    LDX # $30 :- Set raster line.
LP2 CPX $D012 :- Check raster line.
    BNE LP2   :- Is it correct??
    STA $D020 :- Set border colour.
    STA $D021 :- Set paper colour.
    JMP LP0   :- Rerun routine.
```

The reason I disabled the interrupts at the start of the routine is so that the keyboard is not tested, because the keyboard test routine slows down the routine, making it flicker.

The above routine is only to give you an idea of what a raster bar looks like.

You may have noticed that so far, the raster compare register has a wide range of uses, and another one of its uses is to split the screen display.

With the use of split screens, it is posible to have one font on the top line of the screen display, and a different set underneath. The theory of split screens is as follows:-

A. First latch onto the raster line.
B. Set a value in $D018 (where the character pointer is), to point to chr.set 1.
C. Check raster line for second split.
D. Set the pointer value to the second set.
E. Re-run the routine.

If you wanted to find out the raster line that is required, you would use the following formula:-

(LN+5)*8+1:- Where LN is the line number.

In the next issue of CDU, I hope to explain the use of vertical scrolling to obtain the FLD effect. (Flexible Line Distance). All this theory is no use without some practical demonstrations, therefore in the next issue I will provide you with some routines to demonstrate all that we have been talking about so far.

Until we meet again, keep experimenting!

Coball

Joystick poised, fingers at the ready, you are about to fight for your life!

By R. Aldridge and N. Cusworth

'Another pint please Jim'

Alan Withersnips loved his hourly visit to the local pub, 'The Nags Head'. Today seemed just like any other to Alan, as he asked Jim the barman for another pint '...and one extra packet of peanuts purleeeese!!!'

But little did he know.

On opening the peanuts, Alan cried out in surprise as he was sucked molecule by molecule into a fourth-dimensional parallel universe.

Pico-seconds later Alan awoke in rather unfamiliar surroundings. After a short scan of the area, he realised he was, surprise-surprise, in the Starship Coball. A voice rang out in the previously silent environment. 'Hi!!, I'm your friendly onboard computer. Blast those aliens to get lots of points.'

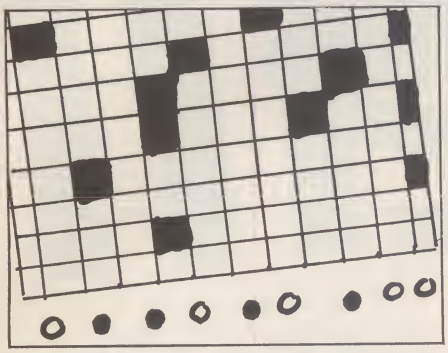
And so began the voyages of Alan the Space Ace, Commander of the Starship 'Coball', ready for anything, anytime, anyplace.

You can plug your vector manipulators into either port.

MICKMON

Get to grips with your programming techniques with this powerful ML Monitor

By Mike Gregory



MICKMON is a Machine Language (ML) Monitor designed for use on a Commodore 64 with 1541 or compatible disk drive. In addition to monitoring the computer's own internal state, *MICKMON* has commands that access the disk drive's memory, and can even address the surface of the disk itself. It also has powerful extras not found on other Commodore 64 monitors, including a 'W' command that can walk through SEI and BRK instructions, allowing investigation of previously unfathomable computer lock-ups.

Using a monitor of *MICKMON'S* power carries its own responsibility. Beginners should remove unprotected disks from their drives before experimenting. All users are cautioned to make back-ups of important disks before using the disk commands. Finally, no matter how experienced you become, you should always experiment with a 'nibble copy' before altering an original commercial disk.

In spite of its power and complexity, *MICKMONS* HELP facility (the '?' command), makes it easy to use by anyone learning Machine Language. Beginners are advised to acquire some experience using 'normal' monitor commands (as described in their ML text books) before directing commands at their disks or disk drives. 'L', 'S', and any DOS commands already mastered may be treated as exceptions to this rule, although even these commands have been considerably enhanced in *MICKMON*.

A beginner's text, suitable for use with *MICKMON*, is **Jim Butterfield's** *"Machine Language for the Commodore 64 and other Commodore Computers"* (Prentice-Hall).

MICKMON loads to $E000-$FFFF, "under" the KERNAL ROM. Once *MICKMON* has been loaded, you can start (or restart) it without recourse to the CDU Disk Menu. Just switch out the Kernal:

POKE 1,52 or POKE1,1

The computer "crash dives" into the Monitor as the ROM momentarily vanishes from its address space and leaves *MICKMON* to handle the next IRQ.

Commodore 128 users will need to be in 64 mode to use *MICKMON*. If you are using a 1570 or 1571 disk drive with your 128, hold down the Commodore key as you poke up or reset directly to 64 mode. *DO NOT* use 'GO 64' as this may not completely re-define your drive as a 1541. Strange side-effects, such as re-formatting both sides of your flippy when you thought you only had access to one, can occur. (You didn't really use side 2 to back up side 1, did you?) These drives should cause no problems when used with a Commodore 64.

Rather than dwell on the fundamentals of ML monitors, we will skip ahead for a "hands on" look at two of *MICKMONS* more advanced facilities — its GD and BW ("GOTO Drive" and "Block Write") commands. In the process, we will also pick up some tips on *MICKMON* usage that may not be apparent from the '?' command's help messages.

The fact that the 6502 chip controlling the 1541 is software compatible with the 64's own microprocessor, the 6510 makes 'GD' a logical and desirable addition to any C64 monitor.To illustrate its use, we will program the 1541 in Machine Language to blink its led light three times on receipt of our 'GD' command.

Before we start, file copy *MICKMON* (the 33-block program) and two or three other files to a newly formatted disk. When finished, reset both the computer and the disk drive by switching the power briefly off, then on again. Make sure your copy disk is the one in the drive.

LOAD "MICKMON",8,1

NEW

POKE 1,1

The screen colour will change as you are presented with a typical ML monitor display. You can change the colours using the function keys F1-F4. When you are happy with the display:

.? (list *MICKMONS* single-letter commands)
.?F (get more information about the 'F' command)

To achieve our goal of blinking the disk drive's activity light, we must toggle bit 3 of address $1C00 in the 1541. Other bits in this register control things like stepping the read/write head across the surface of the disk. If our program contains bugs when we try it, we could end up with the read/write head stuck in a physically extreme location, requiring removal of the disk drive's cover in order to free it by hand. To minimise this potential source of anxiety, we will first assemble and debug the program in the C64's more familiar internal environment.

Inititally, instead of addressing bit 3 of the 1541's Disk Controller Port, we will toggle the corresponding bit in the C64's Border Colour register at $D020. This will simulate the blinking of the led by causing the screen border to flash instead. Once we have the program working correctly we will change '$D020' to '$1C00' then transfer it to the disk drive's memory where it may be executed by the 'GD' command.

.F 7000 77FF

This clears a workspace (to the default byte $00) the same size as the 1541's RAM. (The address range $0000-$07FF corresponding to the 1541's 2K RAM is not generally available on the C64).

.M 7000 707F (to check the result of the previous command)
.+ (repeat until satisfied)

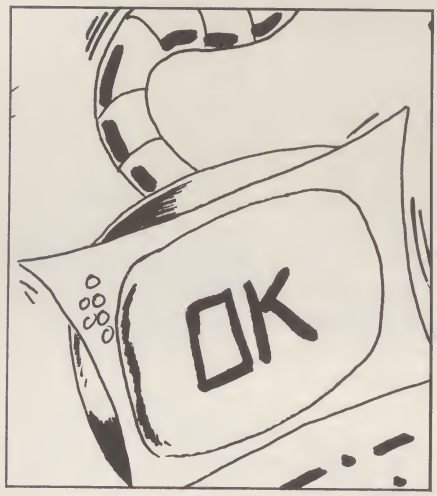
Here's our program. Enter the first line exactly as shown and thenceforth the third and fourth columns only — *MICKMON* automatically supplies each subsequent address. It should look the same on the C64's screen as it does here when you have finished typing. (The comments are optional — include them only if you want a printout).

```
.A 7300 LDA # $06
.A 7302 PHA           ; 6 reversals
.A 7303 LDA # $08     ; (3 blinks)
.A 7305 EOR $D020     ;select bit 3
.A 7308 STA $D020     ;toggle it
.A 730B LDX # $FF     ;(ON or OFF)
.A 730D LDY # $FF     ;
.A 730F DEY           ;delay=$FFFF
.A 7310 BNE $730F
.A 7312 DEX           ;delay=number of
.A 7313 BNE $730F     ; inner loop
.A 7315 SEC           ; interations
.A 7316 PLA
.A 7317 SBC # $01     ;recover count
.A 7319 BNE $7302     ;decrease it
.A 731B BRK           ;next reversal
.A 731C (RETURN)
```

Hit the RETURN key at the '.A 731C' prompt to exit Assemble mode.
F7 will give you a screen dump if your printer is agreeable and online.
If everything else is in order you will see the screen border flash three times when you execute the program:

.G 7300

.D 7300 731B



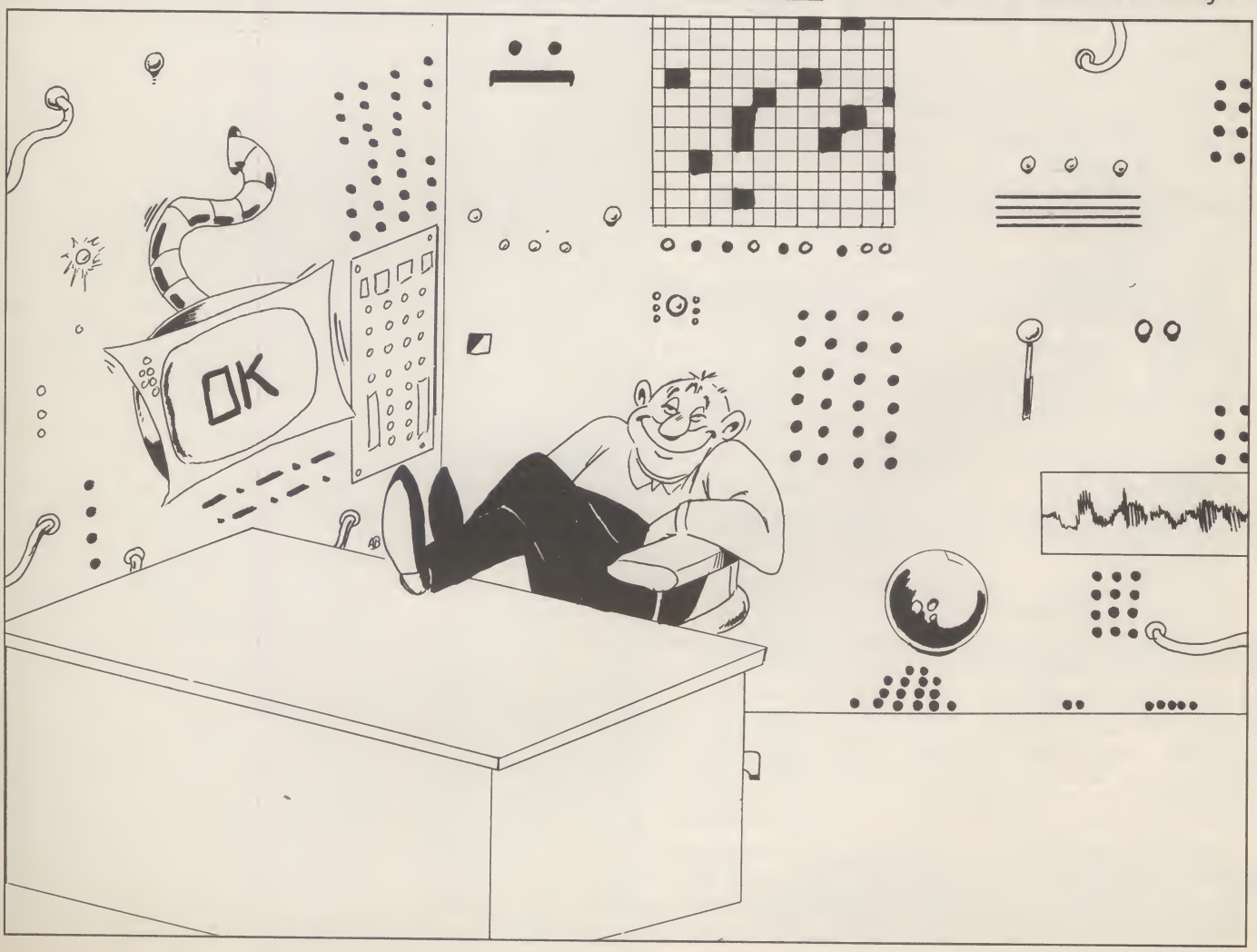
This last step gives a disassembly of the code, showing a variable-length hex dump between the disassembled instructions and their addresses. You may also edit this display, but here, unlike the Assembly listing, you edit the hex code in lieu of the instruction fields.

Our program is going to execute at address $0300 in the 1541, $0300-$06FF being the only RAM normally available for this type of program.

To convert the program to its final form, hit F5 and change '$D020' to '$1C00' as follows:

```
.A 7305 EOR $1C00
.A 7308 STA $1C00
.A 730B (RETURN)
.
.D 7300
```

Now, change the last instruction in the program from 'BRK' to 'RTS'. (Changing the '00' hex code at $731B in the disassembly to '60' will do). Note that code executed with 'G' should always end in 'BRK' if you want to return to *MICKMON*, while 'GD' should only be



used to execute programs terminating with 'RTS'.

Here's what the final program looks like:

```
.,7300 A9 06      LDA # $06
.,7302 48         PHA
.,7303 A9 08      LDA # $08
.,7305 4D 00 1C   EOR $1C00
.,7308 8D 00 1C   STA $1C00
.,730B A2 FF      LDX # $FF
.,730D A0 FF      LDY # $FF
.,730F 88         DEY
.,7310 D0 FD      BNE $730F
.,7312 CA         DEX
.,7313 DO FA      BNE $730F
.,7315 38         SEC
.,7316 68         PLA
.,7317 E9 01      SBC # $01
.,7319 DO E7      BNE $7302
.,731B 60         RTS
```

Although the Branch instruction operands appear incorrect in this disassembly, they use relative addressing and will adjust automatically when the code is relocated. (Provided always that their target addresses are internal to the block being moved). To "see" the program at $0300 (as it will appear to the 1541's microprocessor), mentally substitute '0' for the high digit of any address still starting with '7'. (Do not enter this final "change " into your computer).

Correct any program errors or omissions so that your computer screen matches the published disassembly, then, on a fresh line:

.TD 7300 731B 0300

.GD 0300 (Watch the led light)

To increase the number of flashes:

```
.A 7300 LDA # $14 ;   20 (ten blinks)
.A 7302 (RETURN)
.TD 7300 731B 0300
.GD 0300
```

To make it flash faster:

```
.A 730B LDX   $3F
.A 730D (RETURN)
.TD 7300 731B 0300
.GD 0300
```

If you would like to examine the program after it has executed in the disk drive:

.TC 0300 031B 7400

Then examine it visually (notice how the branch instructions' target addresses adjust to the program's new location):

.D 7400

or 'C 7400 741B 7300' for any list of any mismatches — there shouldn't be any. Change the 7300 to 7200 and try again to see what an unsuccessful result looks like.

The example program was chosen to illustrate a variety of *MICKMON* commands in action. More serious attempts at re-programming the disk drive would typically involve disk access using the 1541's ROM routines. For complete information on the 1541 Operating System and Commodore disk formats, see *"Inside Commodore DOS"* by **Richard Immers** and **Gerald Neufeld** (Prentice-hall). This book must be considered essential reading for anyone hoping to get the most out of *MICKMON*.

Another feature of *MICKMON* is its fast self-relocatablility. Apart from its initial load address "under the ROM", which it continues to use as a RAM-disk containing its own commands, *MICKMON'S* operation is relocatable. The command provides a neat solution to the necessity for a series of monitors which operate at different memory locations.

The 1K "supervisor module", which *MICKMON* initially sets up at $8000, can be relocated by the '*' command anywhere in visible RAM, provided it doesn't clash with the C64's operating system. Individual commands are fetched to this module from under the ROM prior to execution. The longest command needs another 1K making 2K in all. Generally RAM from $2000-$9800 or $C000-$C800 will provide safe location points.

The concept of relocation is designed to avoid memory conflicts between *MICKMON* and the program it is being used to monitor. If a machine language program's load address is $8000 for example you could issue the command '* 7000' from *MICKMON* prior to loading the program. Since *MICKMON'S* start-up code under the ROM is affected by this command, you may even allow the other program to boot normally before using a reset button to "freeze" it in memory. **'POKE 1,1'** will restart *MICKMON* at its new location, leaving the program at $8000 intact and ready for investigation.

You are forced to come to terms with this type of memory management when using the BR and BW ("Block Read" and "Block Write") commands. In order to use them, you must specify the address of the 256 bytes of C64 memory needed to hold the nominated block of information from the disk while it is being operated on by other commands.

.BR 12 01 9100

This tells *MICKMON* to copy track 18 (hex '12'), sector 1 (normally the first block of the directory) into the C64's RAM starting at $9100. This command's format makes it ideal for gathering files into contiguous memory, regardless of how the data is distributed on the disk.

Try out the above command, then save a spare copy of the block to $C100 as a temporary backup:

.BR 12 01 9100
.BR 12 01 C100 or **.T 9100 91FF C100**

For an ASCII dump of all the printable characters in this block:

.MA 9100 91FF

You should recognise the text on display as filenames from the disk's directory. You can edit this text. The dots represent unprintable, though often very important, information. If *"MICKMON"* is one of the filenames on display, overtype it with "NICK-NAME" and copy the edited block back to the disk. (SHIFT/HOME provides a fast way to return the cursor to the bottom of the screen).

.BW 12 01 9100

Is the error light blinking? If so, perhaps you write-protected the disk.

.$E will let you know the Error Status (and stop the blinking).
Now list the directory in the normal way to verify the change:

.$

This is a trivial example of track and sector editing. The real power to be gained from editing a disk comes from changing the numeric information "hidden behind the dots" in the 'MA' listing.

.M 9100 917F

This will show you a hex dump of all the information in the first half of the first directory block. If there are more than eight entries in the directory (requiring more than one block to store them) the first two bytes will represent the track and sector of the following block (usually track 18, sector 4). Since there is no track zero, a first byte of '00' indicates that we are looking at the block containing the end of the directory.

Editing this type of 'structural information' can dramatically alter the 1541's assessment of the disk.

.M 9100 again for convenience, cursor back up and edit the first two bytes of the block so that they read:

.:9100 12 01.....and hit return, then

.BW 12 01 9100 (put it back on the disk)

.$ (display the directory)

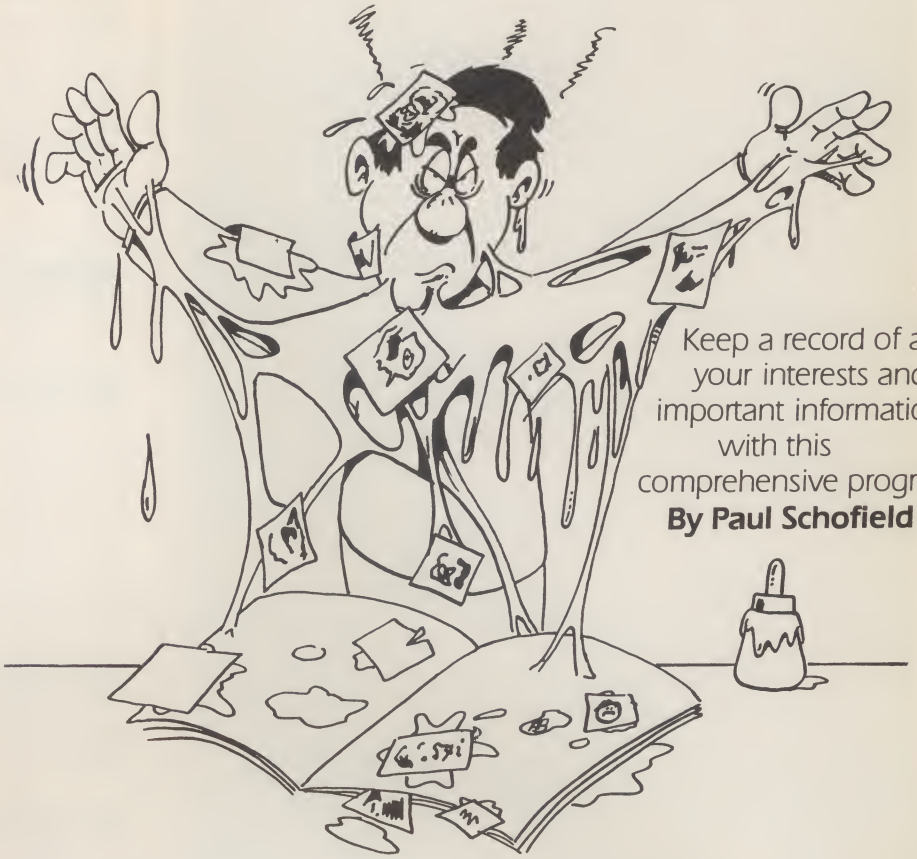
The directory will repeat endlessly because the DOS can no longer recognize its end — finding a pointer back to the start instead. The STOP key will rescue us from this television style programming.

.BW 12 01 C100

That should restore the directory to its original form.

.$ (check)

To explore all *MICKMON'S* features in such detail would require a book. That's what they did for *EASYSCRIPT* and — be honest — did you ever finish reading it? The best way to learn about *MICKMON* is to start using it. If I get enough requests I might be persuaded to write another chapter, but never a whole book. While I was writing the book, you'd be using *MICKMON*, and by the time I had finished, you'd know more about it than me!



Keep a record of all your interests and important information with this comprehensive program

By Paul Schofield

Scrapbook

SCRAPBOOK is a type of database program and as its name suggests it is used rather like the scrapbook you probably had as a child. This may not sound particularly useful, but for many applications it has advantages over the traditional database organisation.

My wife gave me the initial idea for the program when I was teaching her how to use *Fleet Filer*. This is one of the fastest databases available for the C128 and is interesting in that it dynamically allocates data storage like a spreadsheet program. This means it is only necessary to identify the fields of a record as either string or numeric and it is not necessary to define the size of fields. Despite this, for many applications it is still difficult to define a suitable fixed record format.

To take a very simple example of address lists, which are probably the most common database application, how many strings should be allocated for the address? We still maintain separate files for Swiss addresses, that are generally very short and overseas addresses. Despite this some UK addresses still cause problems. Other household applications, recipes for example, are extremely difficult to store in this type of database because of the variable number of ingredients and steps of preparation. The *Fleet Filer* technique does not penalise you heavily on storage for defining records with very large numbers of text fields, but it is still difficult to define the records initially and data entry can be somewhat tedious with large numbers of unused fields.

For such applications, it is more convenient to have a completely free format structure to allow the variable quantities of data to be presented in a convenient format. Features such as sorting and extraction of selected information then become impractical, but are not generally important, provided it is simple to locate the required information. These are the objectives of the *SCRAPBOOK* program.

Organisation of a Scrapbook

Each Scrapbook occupies a full disk less about 5 blocks and has a title to identify the nature of its contents. It also has a password to restrict access to authorised users.

They are best considered as having 9 pages onto which up to 20 scraps of information may be pasted. There is also a unique number (1-180) associated with each scrap, which is used for reference purposes. Logically, page 1 contains scraps 1-20, page 2 scraps 21-40, etc..

Each scrap is a screen display of 20 lines of 40 characters. These may be in either UPPER/lower case (a TEXT Scrap) or UPPER case/GRAPHICs (a GRAPHIC Scrap). Frequently, a single scrap may not be large enough to hold all the information on a particular line. In this case a series of scraps can be chained together by using forward and backward links.

Let's consider the recipe scrapbook example and see how the facility might be used. There are basically two ways in which we may wish to store recipes. If the recipe comes from a cookbook you own, it is likely that only a brief description of the meal, a list of the main ingredients and a book reference are required, but if the recipe is from a magazine full instructions will be needed. In the latter case it is likely that more than one scrap will be needed.

The first decision is the scope of the scrapbook. It could be a compendium of all your favourite recipes, but it is probably more useful to make several more specific scrapbooks, e.g. Indian Cookery, Fish Dishes, Party Snacks etc... It is not essential to make an index scrap as *SCRAPBOOK* provides a directory, which is easily browsed, but in such applications an index can be used to speed up searches. Typically the Indian Cookery scrap book might use SCRAP 1 as a general category index giving a starting scrap for Beef, Chicken, Vegetable dishes etc... The next pointer could be used to go directly to the first category and previous to the last. This information can of course be included within the text of the scrap.

Assume we have used this mechanism to jump to Beef dishes and we view the first scrap for Keema (dry mince curry). A reference to a recipe

book is not possible, so we will probably use more than one scrap. The first to give a general description, to give suggestions for accompanying dishes and notes on preparation and cooking times. The second to list the ingredients and a third for the cooking instructions. Obviously it would be handy to tie the three together using the next and previous pointers and the two pointers free at either end to select the next and previous recipes within the category.

It is not necessary to plan the chaining in advance as it can be changed quite simply at any time. By default all chains point to the current directory page. You should now have a reasonable idea of the structure of a scrapbook so let's start the program.

Starting Scrapbook

To load the program, type **LOAD "SCRAPBOOK",8** this will be from outside the menu.

Alternatively you can load via the menu.

The load takes place in two parts, firstly the run time executive and then application code, so be patient. When it is finished, the main menu is displayed and you are given three options:

- Create a new scrapbook,
- Load an existing scrapbook,
- Exit

To get started you must press **F1** to create a new scrapbook. This I'm afraid is a somewhat tedious procedure as the Commodore Disk Drives are not only exceedingly slow, but also stupid enough to physically write several thousand empty records to the disk rather than simply recording where they will be. You are therefore recommended to have a drink when the program suggests, as it is a good many minutes before the disk is correctly initialised. To identify the scrapbook you are required to enter a title and a 6 character password. The password is to prevent other people from reading and changing your scrapbooks and must be applied before a scrapbook can be loaded, so don't forget it!

After initialising or loading a scrapbook the directory will be displayed for page one. The directory entries contain the following information:

- Scrap number, used to reference the scrap.
- Status, which can be
  - **F** = free for use
  - **T** = text scrap
  - **G** = graphic scrap
- Scrap title, short description of content
- Previous scrap in chain
- Next scrap in chain

The directory level commands are all activated directly by function key as follows:

F1 – NEXT PAGE
Displays next page of directory.

F2 – PREVIOUS PAGE
Display previous page of directory.

F3 – VIEW SCRAP
Prompts for the number of a scrap, which is then displayed.

F4 – EDIT SCRAP
Prompts for a scrap number, and loads it to the Scrap Editor. The Scrap must be on the current directory page. After modifying the scrap text, the next and previous scrap pointers and scrap title can also be changed.

F5 – PRINT
Print this page of the scrapbook directory.

F6 – INSERT SCRAP
Finds the first free scrap on the page, if any. Prompts for the title of the scrap and enters the Scrap Editor with an empty scrap.

F7 – EXIT
Exit to Main Menu.

F8 – DELETE SCRAP
Prompts for a Scrap Number, which is then deleted from the directory. It is only possible to delete scraps on the current page.

NOTE: Any command that prompts for a scrap number may be aborted by entering scrap number 0.

Scrap Editing

Scrap editing is largely a matter of typing in the text you require. At the bottom of the screen is a small help area, which identifies the function keys for HELP, CLEAR scrap, SAVE and exit and flags for text/graphic mode, capslock and insert mode. Selecting HELP provides a help display of the full set of editor commands.

Scrap Viewing

In view mode the following function keys are available:

F1 – display previous scrap in chain

F3 – display next scrap in chain

F5 – print scrap

F7 – exit to directory level

Printer Problems

It was no great surprise that 95% of the problems found in the beta test version of *SCRAPBOOK* concerned the printout functions. If your printer is device # 4, you should get a printer output. In the earlier version, double size print was the default as this looks nicer for 40 column printouts. Unfortunately, it seems that this control code causes problems with some daisy-wheel printers, so it has been taken out. If you have a Commodore compatible matrix printer or intelligent interface to an *Epson* compatible printer, you can obtain double width print by executing the following BASIC commands before running the program.

```
OPEN 4,4
PRINT # 4, CHR$(14)
CLOSE 4
```

If your printer is not device # 4 then load the program, but before entering RUN, enter:

```
POKE 3573, dev      ($0DF5)
```

If this does not work, I'm afraid that you will have to console yourself with the fact that the idea of electronic information systems is to reduce the quantity of printed paper.

Technical Details

The internal structure of a scrapbook is a single relative file with a record length of 40 characters. It is therefore possible to write other programs to access it. The file layout is as follows:

REC. no	Content
1	Scrapbook header
2-181	Directory records 1-180
182-203	Scrap 1
204-	etc

Cellrator

Time is running out! Earth needs its vital Uranium. Can you find enough to survive!

By A. Darnell

The year is 2090, Uranium is running short, but is needed for fuel to power generators to produce oxygen.

The only place where this can be found is in the caves below the Earth, the labyrinth of caves are closely protected by alien life forms.

You are the chosen one, who has the task of collecting Uranium from each cave as you pass through. This can be done by firing your Astro-phiseray at them and transporting them on to your ship *'Cellrator'*.

As you collect the Uranium it will be registered on your items scale. Beware of the cave walls, the enemy have sensitised them, touching them will severely damage your power shields.

No map has ever been produced revealing the way through the caves. The aliens have invisible barriers which constantly change the directions to confuse the intruder.

If you succeed in penetrating the caves you will find the enemy head-quarters. This you must destroy.

Good luck with the mission (You will need it!) and take care of your ship *Cellrator*.

A scrap of paper has been found from a previous mission, on it is scribbled this note: The Uranium capsules need to be shot at different distances from your ship. This also applies to the Gas clouds...ends!!

Rainbow Chaser

Help the Dragon to escape from his shell prison

By K. Murphy

Eggbugg, a little dragon, has had the misfortune of bumping into an evil witch whilst walking through the forest one day. As a result, he was zapped into an eggshell and banished to a far off land for eternity.

When Eggbugg arrives in this new land he manages to find a hippy wizard (Who's heavily into Saxon music), who says he can help Eggbugg escape from the shell and return home.

Before he can do this though, he requires certain items to be collected. These are: A ring of pearls, bottles of potion, golden goblet, Wiz book, large and small crystal balls and about 55 Rainbows.

Eggbugg's task therefore, is to search the land for these items whilst avoiding all the nasties (Not an easy task!).

At the start of the game Eggbugg is given 20 stars for shooting and full stamina. Each time he shoots, his supply of stars depletes and can be replenished by collecting small stars dotted about the land. His stamina can be replenished by collecting small hearts that he may come across. Other collectable items will increase his score.

When he collects a Rainbow, a small counter clicks up until it reaches 20, Eggbugg must at this point return to the Wizard where the Rainbows are converted into points. When all of the items and at least 55 Rainbows have been collected, Eggbugg must return once more to the Wizard where he is awarded a bonus score and a game completed sequence is activated.

Full game instructions are included on the title screen.

Hidden Graphics

Basic programmers can now learn how to use the potential power of the 64 for graphics

By Mike Benn

If the Commodore 64 has REALLY got 64000 bytes, how come the start up screen NEVER has more than 38911 bytes available? The BASIC user is permanently short changed and denied access to almost half the memory capacity he or she has paid for. Somewhere in the machinery is a large block of potentially useful memory, denied to all except machine code programers with massive I.Q.'s.

The operating system and BASIC interpreter hide away large lumps of useful RAM to which only machine code programers can get access. The reason for this rather odd state of affairs, is an addressing problem.

The system designers realised that with an 8 BIT machine the most memory locations you can have is 65536. The computer can therefore access any memory location easily within the 65536 bytes available. What makes the RAM areas inaccessible is the ROM chips riding piggyback on some of the RAM memory, due to the limited location addresses. This ROM memory can be switched in and out, releasing the hidden memory underneath, but by doing so, the BASIC interpreter (or operating system) stops working. No BASIC interpreter and your BASIC program can't run. A fine example of catch 22.

Using machine code programming it's possible to unlock this hidden memory and put it to some practical use. It is ideally suited to various forms of data storage where you need to dip in and out for a block of information. It must be remembered that if you switch off, say, the BASIC interpreter, you've got to switch it on again before your BASIC program can continue.

Hidden Graphics 1 makes full use of this vast, unused memory, storing up to 19 full sized screens at one time but leaving the entire 38911 bytes of BASIC memory available and untouched.

Using the program

The program works with screens using character graphics in hires or multicolour mode. You can use the inbuilt character set or design your own user graphics. I would strongly recommend **Tony Crowther's** 3 into 1 *GRAPHICS EDITOR* for designing both characters and screens or any good character graphics and screen editor will do.

The program stores the individual screens in blocks and are called using the base address of each block. The base address can be located anywhere in the computer but you may get some odd effects.

Areas to be avoided are locations 0 to 2048 and 52720 to 57344. The computer needs some memory for its own use to play and our program needs some.

The SYS calls

SYS 52000, "SCREEN NAME", DV, LA

The first SYS call is a load relocator. This will redirect a previously saved screen to a new location address. This is necessary as individually designed screens are often saved and reloaded to the same address.

DEV = The device number
LA = Load Address to which a screen is to be loaded. Recommended screen locations are as follows:-

SCREEN 1 = 40960
SCREEN 2 = 41960
SCREEN 3 = 49260
SCREEN 4 = 43960
SCREEN 5 = 44960
SCREEN 6 = 45960
SCREEN 7 = 46960
SCREEN 8 = 47960
SCREEN 9 = 48960
SCREEN 10 = 49960
SCREEN 11 = 50960
SCREEN 12 = 57344
SCREEN 13 = 58344
SCREEN 14 = 59344
SCREEN 15 = 60344
SCREEN 16 = 61344
SCREEN 17 = 62344
SCREEN 18 = 63344
SCREEN 19 = 64344

SYS 52003, LA, MODE, CS, CC, B0, B1, B2, BR

The second SYS call controls a number of variables.

LA = Location Address (This should be the same as the load address for each screen.)

MODE = Character mode (If the value of zero sets HIRES mode, A value one for MULTICOLOUR mode)

CS = Character Set (20 should be the default value for this variable. Consult your screen designer manual if you use your own user defined graphics. If you do use your own character set don't forget to raise the start of BASIC, again check your manual.)

CC = Character Colour (The value can be any figure from 0 to 15 in HIRES but must be between 9 and 15 when used in MULTICOLOUR mode.)

B0 = Background colour 0 (Values = 0 to 15)

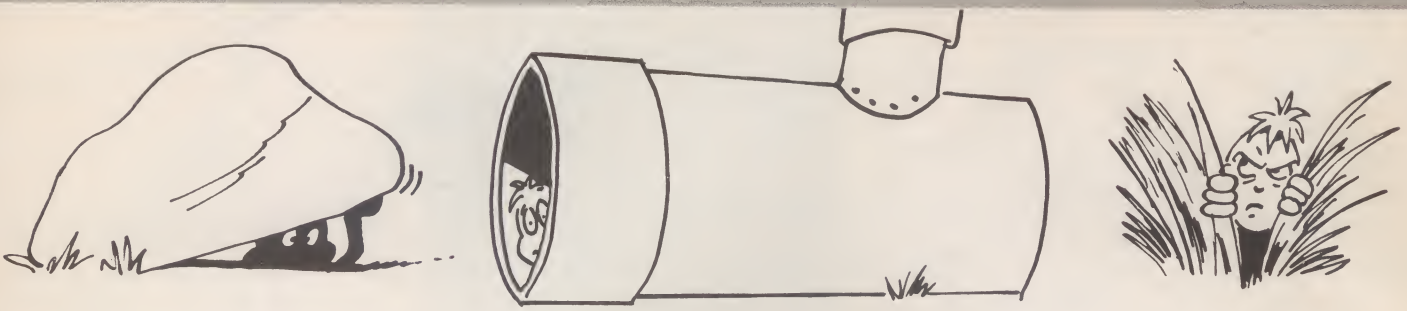
B1 = Background Colour 1 (Values = 0 to 15)

B2 = Background colour 2 (Values = 0 to 15)

BR = Border colour (Values = 0 to 15)

SYS 52006

This third SYS call clears the screen and returns you back to BASIC programing mode.



Running Hidden Graphics 1

From the 'Menu' select the program *Hidden Graphics 1.* This will load the Basic demonstration program, which in turn will automatically load the M/C which is saved as HG1.code.

To run the Demo program from outside the menu program, simply type 'LOAD "HIDDEN GRAPHICS 1",8 then RUN.

Hidden Graphics 2

The program allows for mix and match backgrounds and animated screens. Three dimensional effects (well two and a half) could send you speeding down endless roads in an 'Out Run' clone. Split screen backgrounds can be easily created if your latest megabuster project requires such a use. All you have to do is work out the game play, design the graphics and a little skill at programing. Easy!

Hidden Graphics 2 works by splitting the screen into five zones, each zone being a full screen width and 5 lines in height. By creating blocks of the same proportions they can be placed and swapped into any of the five zones. The use of the blocks is similar to the definition blocks used in sprites. In this case each block of data is 200 bytes long instead of 64 bytes used by sprites. Similarities don't stop there; by storing the blocks one after another the program looks for whatever block is called. If you want to put the third block in the series on the screen then we enter a 3 in the appropriate variable. As each block is called it can be given a separate colour independent of other blocks already on the screen.

Designing the graphics block can be done using any good character and screen designer. The simplest way is to create five blocks at a time on a full sized screen and save a screen as a group of five blocks. Be careful when relocating the loads that they don't over write the machine code program (see details of load areas).

You can of course use your own user defined characters but remember to protect them by raising the BASIC memory to stop them from being over written. The following line will protect any redefined characters and has the advantage of protecting any sprite data.

POKE 43,1: POKE 44, 64: POKE 16384, 0: NEW (Type this line in direct mode)

The down side of raising the start of BASIC is a loss of program space but it's rather that, than having your program and graphics corrupted.

The SYS calls

The following sys calls control the program.

SYS 52600, 'BLOCK NAME', DV, LA

The first SYS call is a load relocator. This will redirect your saved graphics blocks to a new address. This is necessary as your blocks (or screens) will probably have been saved to the same address. To maximise the number of blocks that can be called on, the first block should be loaded 41000 ($A028). As each block is made up to 200 bytes then simple arithmetic will allow you to calculate which address to load the next block. You must avoid loading any block that would over write 52600 ($CD78) to 57399 ($E037) or the program could become corrupted. Memory from 57344 ($E000) to 65536 ($FFFF) can be used to store your graphics.

DV = Device being used

LA = Load Address (See above)

SYS 52603, BA, MD, CS, B0, B1, B2, BR

BA = Base Address (This should have the same value as the start of your graphics block less 200.)

MD = Mode (0 = Hires. 1 = Multicolour)

CS = Character Set (See your graphics editor manual for details)

B0 = Background colour 0

B1 = Background colour 1 (Multicolour 1)

B2 = Background colour 2 (Multicolour 2)

BR = Border colour

(**NB.** A syntax error will be caused when working in hires mode, if the program fails to find a value in Background 1 and 2. The values will have no effect in hires mode.)

SYS 52606, SP, BD, CC

SP = Screen Position (This screen has been divided into 5 zones running down the screen. A value of 0 places a block at the top of the screen and a value of 4 at the bottom. Do not use a value higher than 4, there is no error trap)

BD = Block Definition (The memory available allows for the storage of 79 different blocks, the first block being called with 1 and so on. If the suggested base address is used then definition numbers 59 to 81 or random data will be loaded to the screen.)

CL = Character Colour (A blocks individual colour.)

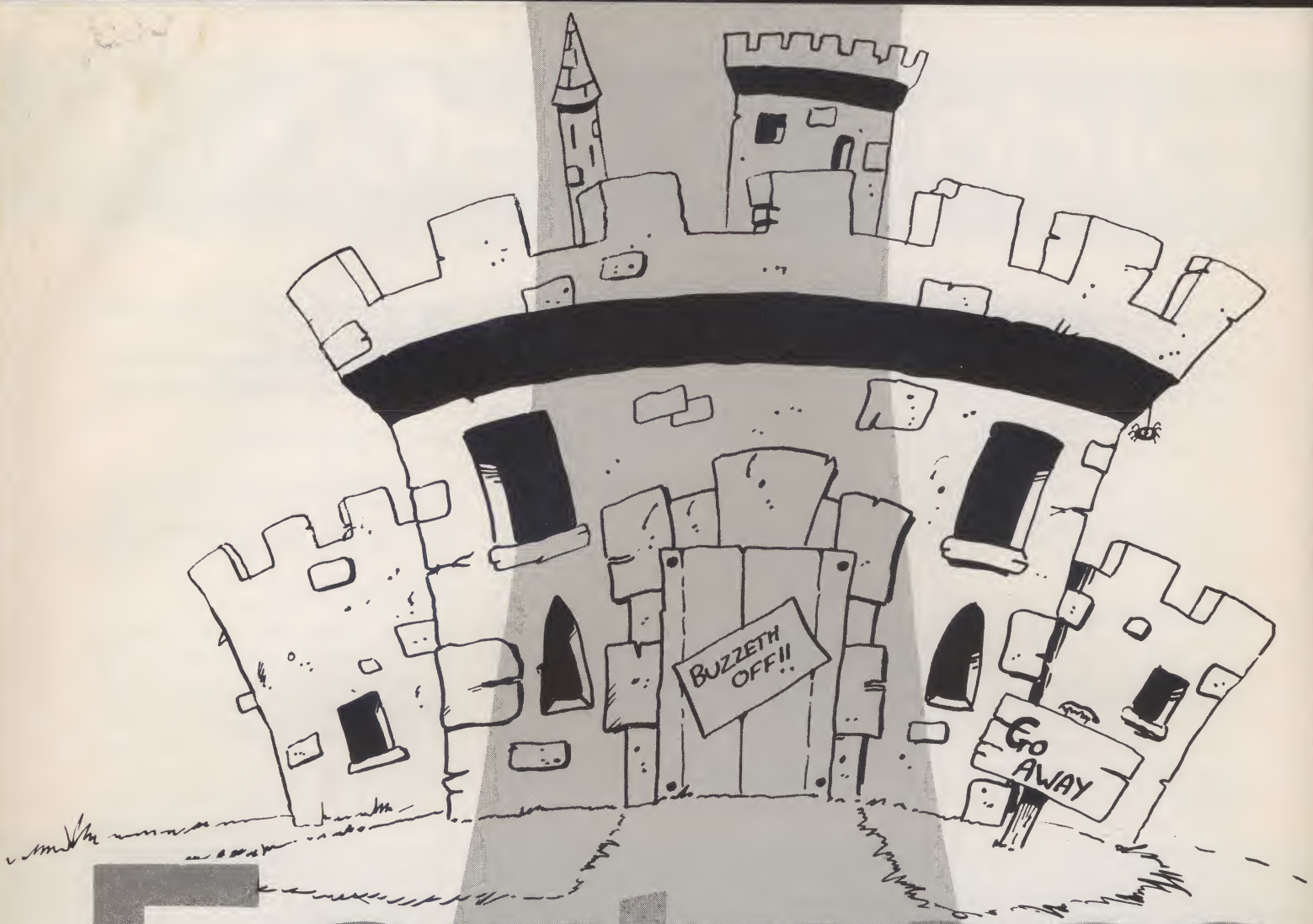
SYS 52612

This call clears and resets the screen.

Running Hidden Graphics 2

From the 'Menu' select the program *Hidden Graphics 2.* This will load the Basic demonstration program, which in turn will automatically load the M/C necessary for the demonstration.

To run the Demo program from outside the menu program simply type 'LOAD "HIDDEN GRAPHICS 2", 8 then RUN.



Fortress

Put your flying skills to good use in this blast 'em game

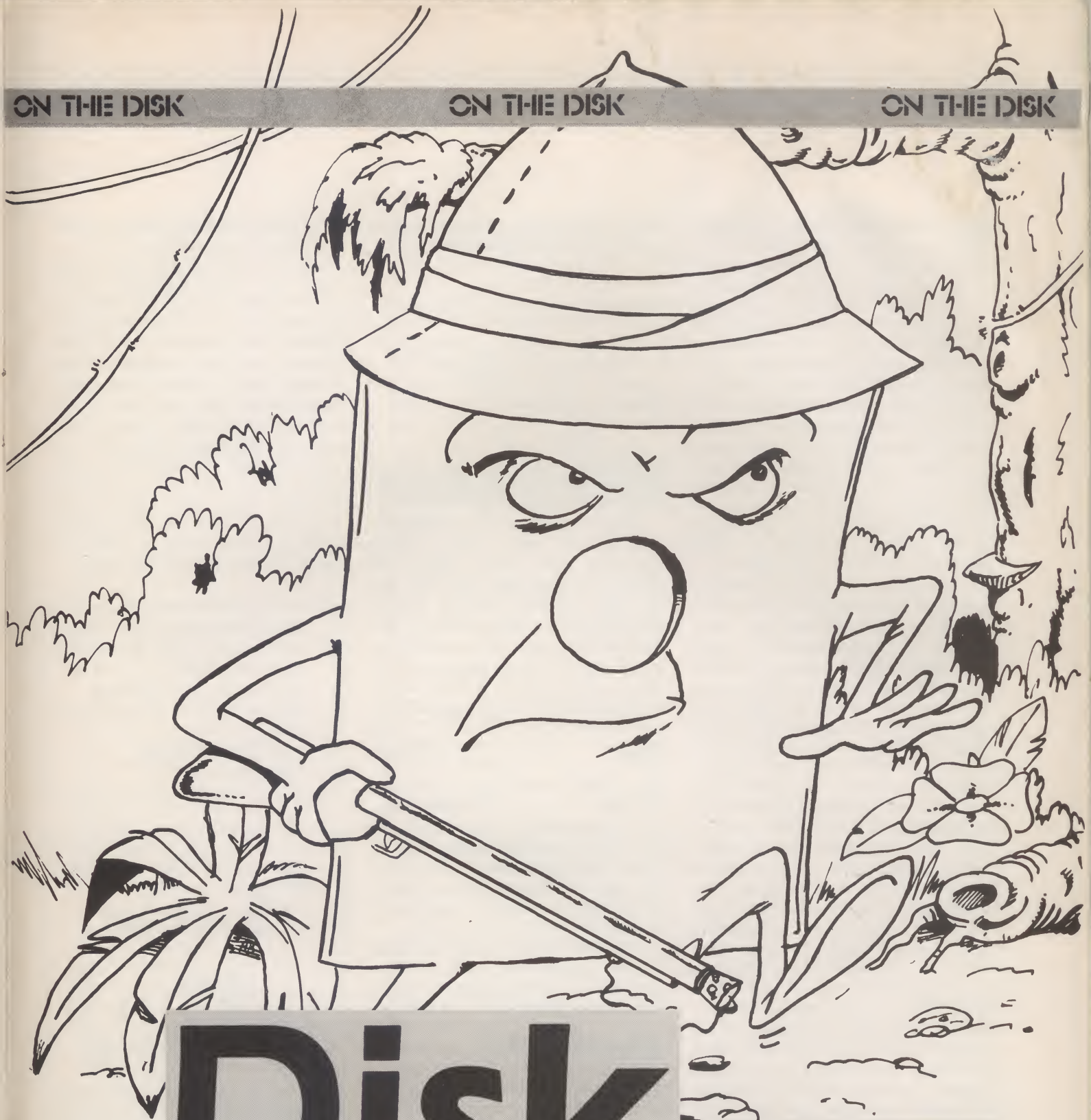
By G. Mayhew

The object of the game is to guide your hypercraft through the enemy fortresses (6 in all) and reach the teleport at the end of each.

Whilst flying through the fortress there are some obstacles which have to be eliminated. To do this fire your laser at the blocks which will then disappear.

There are also several small gaps which your ship must manoeuvre through without touching any of the side walls, as this will make the fortress unstable, resulting in the demise of your ship and crew.

An extra ship will be gained on the 3rd level, and if all 6 are completed the speed at which the fortress moves is doubled, making your task even greater!! The RESTORE key will abort the game in progress at any time, RUN STOP will pause the game until fire is pressed.



Disk Hunter

Find those elusive disks easily with this library archiver

By Mike Holmes

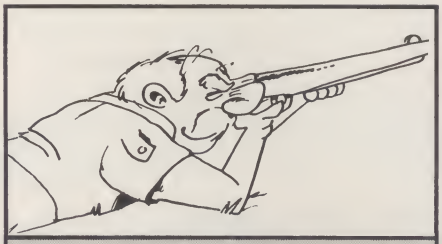
If you have been using a floppy disk drive for any length of time, it is almost certain that you have accumulated a large quantity of disks, not only of your own making but also from other sources. If, like me, you are loath to reformat some of these for re-use just in case they might contain something valuable, then there inevitably comes a time when you cannot be sure what's on what and where, especially if the files become months or years old.

This is a particular problem if you are using your computer for business applications, where finding a file can suddenly become vitally important. This is most likely applicable to word processor files, letters and such.

For instance, if someone makes a reference to an invoice or a letter that you had written six months ago, how do you go about finding it? The usual method is to scroll through successive disk directories in search of the elusive file in question, which is alright if the disks and their contents have been well organised, but as with the best of us, this isn't always the case. Searching disk directories by hand is time consuming, frustrating, and attended by the chance of missing the very file you're trying to find. (With the added hassle of repeatedly swapping disks, which becomes a chore after 30 minutes or so). One could keep printed lists of directories on paper, but I've never managed to do this for all my disks, and in any case it dosn't alter the problem of searching.

The problem is equally true where programs or Basic utilities are concerned. In the dim and distant past you wrote a subroutine which does such-and-such, incorporated in a program whose name you cannot remember accurately, but now you have a new use for it and would like to find it again. The elusive file could be anything from word processor text to numeric data saved by a Basic program to an assembled, loadable machine code routine, but how to find it?

There is a saying which goes 'let the machine do it', since a task like this is ideally suited to a computer, given the necessary software. The program is called *'HUNTER'*, and has a single-minded purpose – to create, update and build upon a library of disk directories, and to search the library when required for the name and ID of the disk containing any file.



Hunter with single disk drive

Before running *Hunter* for the first time, you will need to have a newly formatted disk ready, which will become the library disk. If you like, you can copy *Hunter* to this disk, as long as it is one of the first two files in the library disk directory. The reason for this is that *Hunter* uses the directory of the library disk directly to find out what's on it, but ignores the first two files as one of these might be itself, the *Hunter* program. In this event the library disk will be entirely self-contained including, as it does, the program. The second file could be some other useful utility, or a 'dummy' file to pad out the first two entries and visually separate *Hunter* from the library entries when this disk's directory is listed on screen, for example it could be named '.' or '----------'.

This being done, you can now load *Hunter*. It is an executable machine code file and is loaded and run using the secondary address of 1, as follows:-

```
LOAD "Hunter",8,1
```

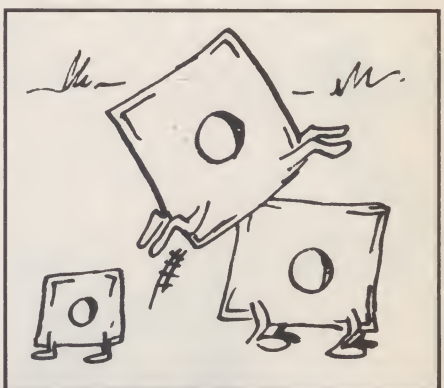
(The above is providing you have used the Copier to save a version of *Hunter* onto another disk, prior to using the program).

As you will have noticed if you have seen *Hunter*'s disk sector count, it is a little on the large side, which will seem a bit strange considering the program apparently doesn't do very much. The menu has only six options, and one of those is for quitting the program.

Because of the time, which to be fair is not that long, taken for *Hunter* to load, you can be doing something else while you are waiting until such time that *Hunter* announces its presence by bleeping at you.

We are now at the menu stage. The brown screen and yellow text is not terribly exciting, it must be admitted, but this is a working utility rather than a graphics demonstrator, and the subtle colours will ensure that the text can be read from your average, battered old black-and-white TV as well as a pin-sharp, quality colour monitor. The program identifies itself with 'C64 DISK FILE HUNTER' in a big, yellow block, followed by a one-line message: 'Single Drive Operation'. This means that all disk operations will be carried out on one drive (device 8). There are six options on the menu:-

1 – Add a disk directory to the library
2 – Search library for the disk(s) containing a file
3 – Delete a file from the library
4 – See the directory of a disk in the drive
5 – View single library file
6 – Quit program



A little diversion

Before any searching can be done there has to be some directories to search through. Therefore option 1 will be the first option you should use to add directories to its library.

The way this is done follows a convention of the C64 in the way that it handles disk directories. Or, to be more accurate, the way that it can't normally handle disk directories in its powered-up-from cold state without any software help, and for that matter

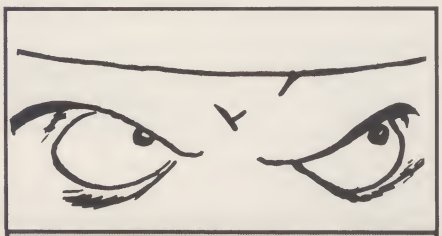
any other sort of DOS operations. Being a tape-based machine, designed at a time when floppy disk drive ownership by the average home user was so exclusive and expensive as to be totally out of the question, the 64 does not know the existence of the disk drive. Well it does, if these are referred to as external devices 8, 9 etc.

Consequently the Commodore disk drive, as an intelligent device, has to go some way to bridge the gap. An example of this is the bizarre manner in which a disk directory has to be got and displayed on screen for examination.

The direct mode command **'LOAD"$",8** is used on the Commodore, and as far as it is concerned it is going to load a Basic program from device 8. The drive on the other hand recognises that "$" as meaning the disk directory. Nowhere in the disk drive or on disk does the directory exist looking like a Basic program. It is generated entirely by the drive's software, using the filenames and their relevant information drawn from the directory track, and sent to the unsuspecting 64. The drive has to first send a two-byte integer which the 64 will store in its 'pointer to start of Basic text' zero page location. Then follows the first contrived Basic line, comprising a two-byte integer as a pointer to where the next Basic line will start in memory. A two-byte integer as a line number, but is actually the zero digit (specifying side '0' of the disk if it were in a double sided drive) which precedes the disk name, then the disk name, ID and DOS format code 2A, and finishing with a zero byte which always ends a Basic line. Each following disk file name is sent in identical manner, where the sector count occurs where a Basic line number would normally be, the file names are put in quotes and the line finished off with one of the file-type names, all of which are made up on the instant according to the value of a single coded byte in the true directory actually written on the disk. The number of bytes free is the last line. The result is produced on screen by LISTing as you would a Basic program. Also, as with a Basic program, the LISTing can be exported to external devices. As a typical example, printing it out on paper would take the form: OPEN4,4,0: CMD4: LIST and here the LISTing is converted into an ASCII text file as it is sent to the printer. But it could just as easily be sent to a disk drive and Saved as a text file on disk. Converting a disk directory into an ASCII file and saving it on disk with exactly the same familiar appearance as it would have when LISTed on screen would follow this sequence of events:

```
LOAD2 "$",8
OPEN2 "8,2, "name for file, W,U"
CMD2
LIST
CLOSE2
```

This then is the way the *Hunter* files its entries.



Using Hunter

Option 1 from *Hunters* menu – you select it by simply pressing the '1' key, – which is immediately followed by a message and a bleep telling you to remove the Hunter/library disk and insert the disk whose directory you want to include. This done, keying RETURN allows *Hunter* to open a channel to the drive like a Basic load, giving the DOS the $ filename. As the directory is received *Hunter* translates it from its Basic format into ASCII lines with carriage returns at the ends, and temporarily stores them. In the process the directory is listed onto the screen simultaneously, so you can see what you're going to get. When it's all got in, a message instructing you to swap the library disk back again follows with a bleep. Again keying RETURN causes *Hunter* to file the new directory into the library. The file name chosen for the new file is contrived automatically from a combination of the disk name and its ID; the two character ID is inserted into the fifteenth and sixteenth characters of the filename. These may overwrite the disk name part if it's very long, but this doesn't matter. The actual filename used for saving, is displayed on screen as *Hunter* the file.

The process is both as fast as the disk drive will allow (as is usual) and easy to do. You can get through a stack of disks in next to no time. Should you want to refile an updated disk at a later date, which already exists in the library, this is no problem for *Hunter*. Detecting an existing file, deleting and rewriting is also automatic. You are now, however, warned about this should it occur, another reason for making sure all your disk ID's are different from one another (Smacked botty if they are not!!) *Hunter* assumes that you know that you are replacing the directory of a disk with the same name and ID. It is important that refiling an updated disk is as fuss free as it was when filing it as new.

Similarly, as it will be a pain to load *Hunter* to file the directory of just one disk if required, this can be done in direct mode in the manner described earlier for LOADing and LISTing the directory to a file of type USR, and the format will be familiar and understandable to *Hunter*. On concluding its file adding/overwriting activities, *Hunter* returns to the start-up menu.

Option 2 from the menu is the one which initiates the searching function. The library disk has to be in the drive for this. You are then asked for the filename to find, or a part description of the filename, and a flashing cursor indicates that *Hunter* is ready to input the text. The text can be edited in the normal way and all printable characters are accepted. However, because the lower case character set is active don't use capitals unless you know the filename to be found was entered in shifted characters! In practice it is likely that *Hunter* will be hunting filenames on a part description only, since this is easiest to do.

Hunting is carried out by first reading the directory of the library disk itself, which is listed on the screen in the process, and this is used to read each directory entry file in sequence, as they appear in this directory. Searching is performed on each line through the list by comparing the strings lexically; the line is scanned

progressively from beginning to end while *Hunter* compares groups of characters in the line with your keyed-in bit of text and to the same length. It is similar to a **'IF A$ + MID$ (B$, X, LEN (A$))'** situation, the important difference being, that it is maybe a hundred times faster than if Basic were doing it. The advantage is that part of a description can be found anywhere in a line.

Not only names of disk files can be hunted, but also sector counts and file types. Because the directory file is simply text, you can search on 'PRG' for instance and list all names of disks containing 'PRG' files. Whenever *Hunter* accesses a directory entry list for reading, it prints the disk name and ID (the first line of the file) of the disk it came from, but this line is never searched. The disk name and ID of the directory about to be read are always printed to screen, regardless of whether any of its 'files' matched your entry or not. Pretty soon the screen fills with a list of inverse video disk names, all in the format of the first line of a directory. If a match occurs in a 'filename' line, that line is also printed, accompanied by a bleep, and others if more matches occur throughout the same disk directory list, and these relate directly to the disk name immediately preceding them. When the list is exhausted *Hunter* gets and searches the next list and so on. The result is a (fairly) slow scrolling display of what appears where and how often. Or not, as the case may be.

Note that at all times, whatever operation is in progress, LISTings can be halted by pressing the STOP key, any other key will cause the LISTing to continue.

Another point to note here is that *Hunter* can be stopped at any time by keying **RUN/STOP & RESTORE**, then **'r'** to re-run from the start.

If, on choosing option 2, you enter nothing for *Hunter* (i.e. carriage return on a blank line) then what you will get is the whole of every list printed on the screen, which can be useful for simply skimming through all the directories. Single characters can be searched, for example, it might be amusing to find out how many filenames have the letter 'x' in them. But otherwise parts of filenames such as 'let' for 'something letter' is more useful. Searches on sector counts can be done by entering digits. For instance, you can't remember what it was called but you are sure it was 9 sectors in length when it was saved. Searching on '9' causes every directory line with a '9' in it to be viewed, including of course, those with sector counts including '9', such as '49', '239' etc etc.

Also, as you may have gathered, since the 'blocks free' last line is included at the end of each line, this can be searched for as well to find out how much free space each disk has. Searching on 'free' will cause the 'blocks free' line of every list to be printed, immediately preceded by its relevant disk identifier.

On finishing with the last list, *Hunter* waits until a carriage return is keyed while you peruse what's on the screen. Keying **RETURN** brings back the menu.

OPTION 3 is used to delete or scratch a file from the directory. An invitation to enter the filename follows and in fact is started off with the characters 'Sø:' preceding the cursor, which will be input as the front end of a command instruction which will be sent to the DOS on keying **RETURN**. These could be rubbed out and replaced with some other command if you wished, like 'I' to initialise the disk or 'Cø:' to copy a file. After sending the command, the DOS error message is printed, and *Hunter* waits until you press any key to return to the menu.

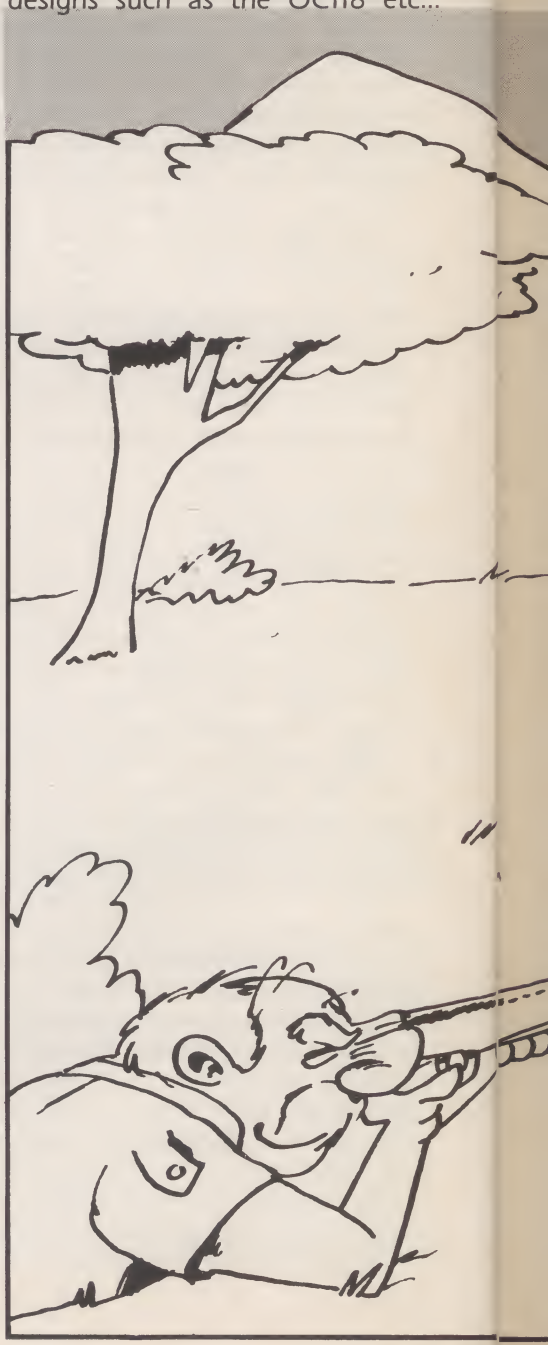
Option 4 lists a directory of a disk in the drive. You can slow the list down by using the CTRL key or use the STOP key. On completion of the list, any key returns you to the main menu.

Option 5 allows you to view a single library file. To do this the library disk must be known first (use option 4 to jog your memory). The function asks for the disk name in library, with a mention that the wildcard '*' should be used afterwards as it is impossible to be able to know how many spaces there may be after the disk name and before the ID included at the end (as filed into library and derived from source disk name and ID). This also means that only the first part of the library disk name to be viewed need be entered, making things a good deal easier and quicker (beware, of course, of similar disk names).

Option 6 quits *Hunter*. The program shuts down and exits in an orderly manner, and resets the 64 with a cold start to Basic.

Hunter with two single disk drives

To be able to use a second disk drive this has to be either 'hard-wired' as device 9, or the DIL switches set for device 9 in the case of alternative designs such as the OC118 etc...



Alternatively, a device changing program or utility run to change one of your drives to device 9.

With the second drive designated as device 9, and a disk containing the *Hunter* program in drive 8, another disk (formatted and containing files, preferably) must be in drive 9, and then *Hunter* can be loaded. There is no need to instruct *Hunter* that drive 9 is active, it will find this out for itself when it runs. A test is performed to see whether device 9 is present, and which also is why a disk has to be installed in it. If *Hunter* cannot open a test channel to the disk it will assume drive 9 is unavailable and resort to single drive operation.

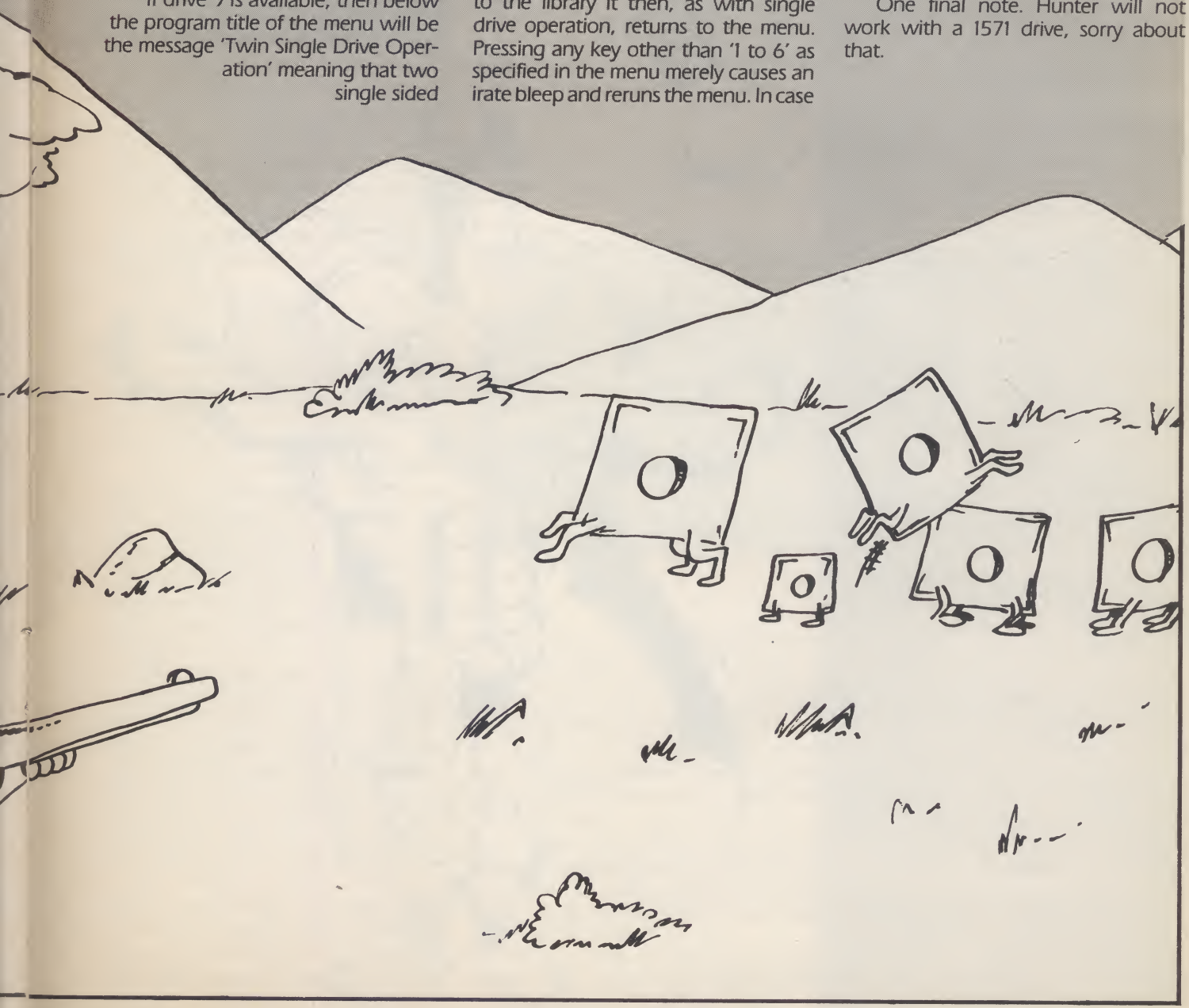
If drive 9 is available, then below the program title of the menu will be the message 'Twin Single Drive Operation' meaning that two single sided drives, 8 and 9, can be used. Actually drive 9 is only used to load disk directories to add to the library disk, so that the latter need never be removed from drive 8. In this event keying '1' from the menu to add a disk directory generates the message 'Is disk to read in drive 9?' key RETURN when ready', whereupon the directory will be read from drive 9 and saved straight onto the library disk in drive 8 with no further pauses. It is handy for creating a library from a load of disks, as only these need be inserted into and removed afterwards from drive 9, while the disk in drive 8 steadily collects all their directories.

When *Hunter* has added a directory to the library it then, as with single drive operation, returns to the menu. Pressing any key other than '1 to 6' as specified in the menu merely causes an irate bleep and reruns the menu. In case of any real difficulties, the RUN/STOP and RESTORE keys can be used to generate an NMI, and the keying '**r**' will restart *Hunter*.

Finally, no other files of any kind must exist on, or be added to the library disk. The program expects everything except the first two entries to be in the familiar list format. This doesn't stop *Hunter* searching through anything it finds though, and although it will happily plough through almost any text file, the screen display will be strange to say the least.

That, then, is about it. Generally if the right disk is in the right drive at the right time you shouldn't have any problems.

One final note. Hunter will not work with a 1571 drive, sorry about that.



Super-file 64

Blow up that filing cabinet!I use Super-file 64 to handle all your filing

By Madhu Surendranath

Welcome to *Super-File 64*, the easy to use data-base program written by **Madhu Surendranath**, age 14.

Instructions for use

Super-File is a menu driven data-base program allowing you to store things as if it were a filing cabinet. It allows up to about 200 records in a file. Here are the features and how to operate them.:-

1) **Create Fields** – (With a maximum of ten fields). If you make a mistake, continue to the end, you will then be given the opportunity of correcting any errors. When you have finished, you will be taken back to the main menu.

2) **Add record to file** – You will be presented with the required amount of fields on the screen. Type in the information for each field. If a field is not required for that record, type a "/ ", this makes saving the file to disk better and safer. Follow the prompts to either return to the main menu or add a record to the file.

3) **View the records** – You will be asked where you want to start viewing from. If you want to abort viewing, type in **'0'** to return back to the main menu.

4) **Disk operations** – In this option, you can load or save previous files, view the disk directory, format disk and other options as well.

5) **Tape operations** – In this option, you can load or save files but in this case to cassettes.

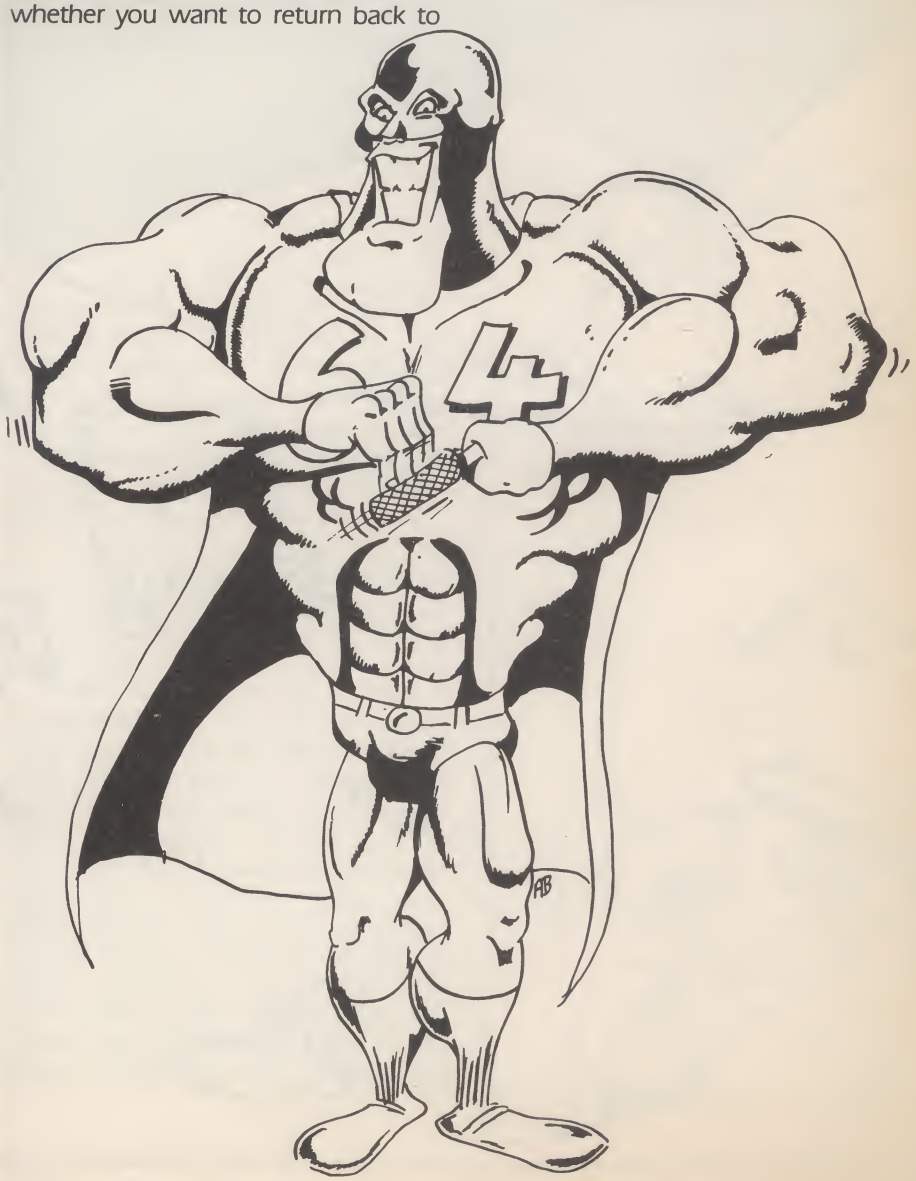
6) **Printout records** – In this option, you can print out selectively or the whole file in either file format or label format, whichever you prefer.

7) **Modify records** – In this option, you have the chance to correct any mistakes you may have made. It will ask you which one to change and after that, just follow the prompts to correct it.

8) **Sort/Search** – In this option, you can sort the files into order choosing what field to use for the sort. Again, please follow the prompts. To search an item, use the prompts to help you. You can again choose what field to use for the search. It's a very useful and powerful feature of this database program.

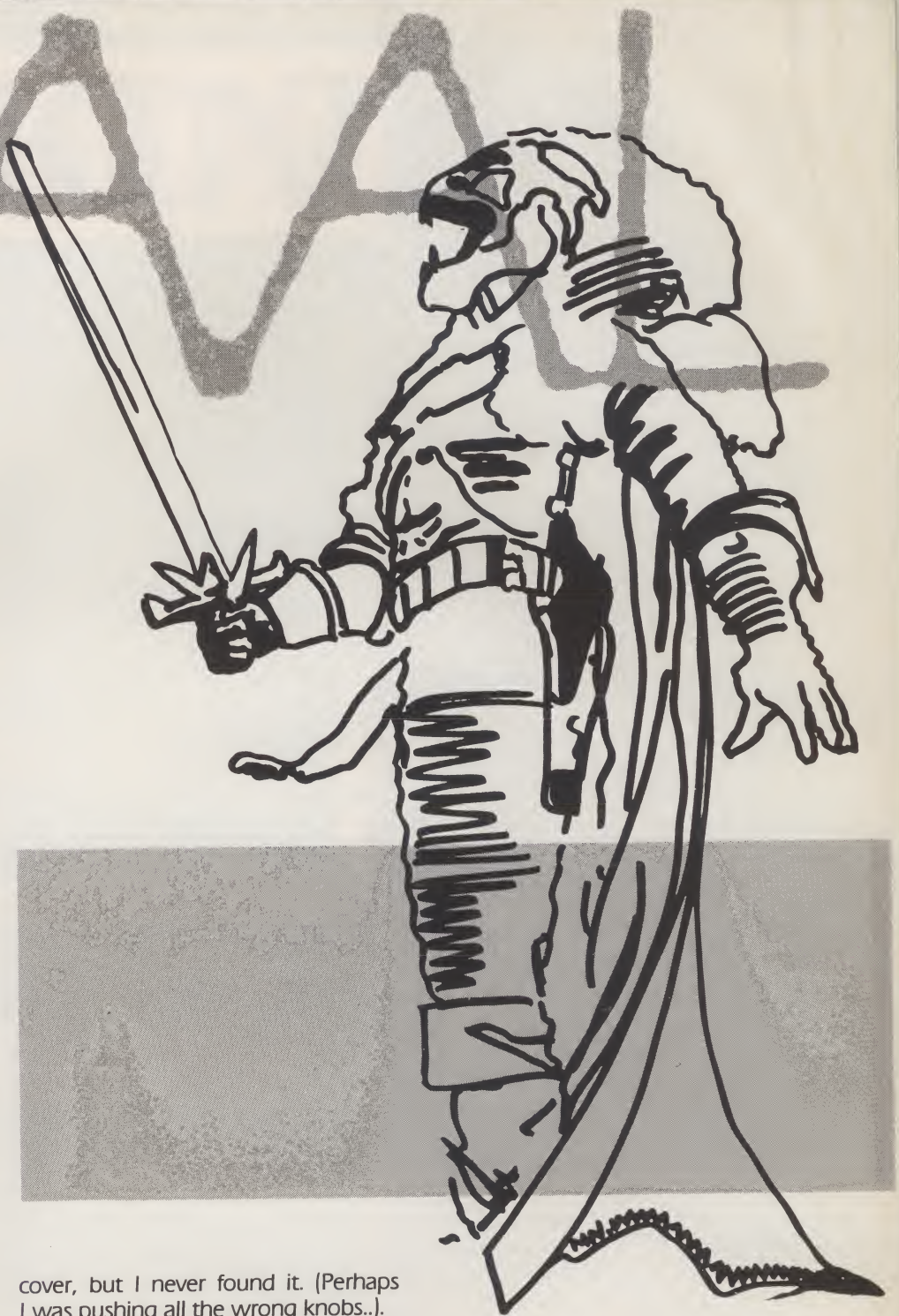
9) **Quit Program** – You will be asked whether you want to return back to *Super-File*.
WARNING: If you reply 'yes' to this question, your data cannot be retrieved unless it has been saved.

Please note that if a printer has not been connected, do not try to use the printer functions. You could lose all your data and the program might crash!



BAAL

Reviews



'The future of the world lies in the hands of an elite squadron of men! YOU are the leader of the Time Warriors. Can you save the earth from the evil BAAL'? Surely not THE Baal?' Come on men, let's go.....

Your aim is to find all the pieces of the war machine, and waste Baal in the process. The pieces are scattered amongst 4 domains: 2 large and 2 small. These domains are built up of platforms, ladders, transporters and mines (Argh!). Once the required number of machine pieces are found on each level, you can progress to the next by finding the teleport station and smashing the fire button.

So! Out come the lasers, set to power mark one, the weakest, (with 3 other settings available if the cartridges are found). Should your laser energy, (Electrolite), get too low, the fired will glow, so it's off to the local BP refuelling point for a recharge.

Should you come across one of the 100 aliens, or 400 traps and fail to blast them, one of your five lives will be lost. (you are then beamed down to the nearest 'Beaming Point'). Don't be too disheartened mateys! a new warrior is designated to your squadron every 5000 personal status (!?) points you achieve.

The graphics are very colourful, ranging from brown cavern type scenery to blue bricks and golden temples and statues. Sprite graphics are very varied and crisp. Some of the aliens are really quite funny, a blue bouncing frog being one of my favourites, Sound is limited to 'Chink, chink, chink' as you run across the floor, and 'Boom' when an alien meets its creator, which is a bit of a shame really. One consolation is the heart stopping 'Blam' when you blow up in mid flight. This is amazing and really made me jump up and out of my skin. Music is mentioned on the cover, but I never found it. (Perhaps I was pushing all the wrong knobs..).

As for the game play. I found it a little too difficult, and the game will certainly need a lot of play before any amount of levels are completed. I soon found myself looking for my expert cartridge and location $???? (Hehe) for unlimited lives.

I found one small bug, when I jumped to my death off a platform into a transformer, I died, transformed, died, flew out of the transformer and died again. Most strange!

All in all, this is a hard game with a lot in it. If you like your games hard 'n large, go get it. If not, try it first.

At a glance

Title: Baal
Suppliers: PSYGNOSIS, PORT OF LIVERPOOL BUILDING, PIER HEAD. L3 1BY. TEL: 051 709 5755
Price: £12.99 DISK £9.99 TAPE
Graphics: Colourful and varied
Sound: A little disappointing
Playability: Quite difficult to master
Addictiveness: Once mastered, it'll drive you mad

VIDEO TIT

Imagine you are preparing a video presentation for a school or college project or similar, and you want to add some kind of text screen to it in the way of explanation, title etc. You could feasibly make up some cards with text on them and video them. OK it works, but it isn't very professional is it? A method of inputting text screens directly on to tape would be better. Actual animation of the graphics would be superb!

Enter, *The Video Title Shop* from **Datasoft**. Supplied on two 5.25 inch disks along with a 50-odd page manual and a couple of other bits and bobs, this package is capable of producing impressive title screens. Along with one, or preferably two video recorders, the *Video Title Shop* (VTS) may be used as described above.

VTS is actually only one of the programs supplied on disk. Also on the disk is *Micro Painter Plus* (MPP) which is basically a graphics picture editor. It may be used to produce backgrounds (called canvases) for use with VTS. Several pre-programmed canvases are supplied on the second disk to get you started.

You can use VTS to produce static title screens for video presentation, but it can do much more than that. A block of text on the screen is known as an object. Objects may be acted on by one of a number of effects. More about this in a moment, but first, a bit more theory. A title sequence is made up of one or more pages. On each page there may be several objects which may themselves be acted upon by effects. Certain effects act upon the page as a whole, rather than on the objects on it.

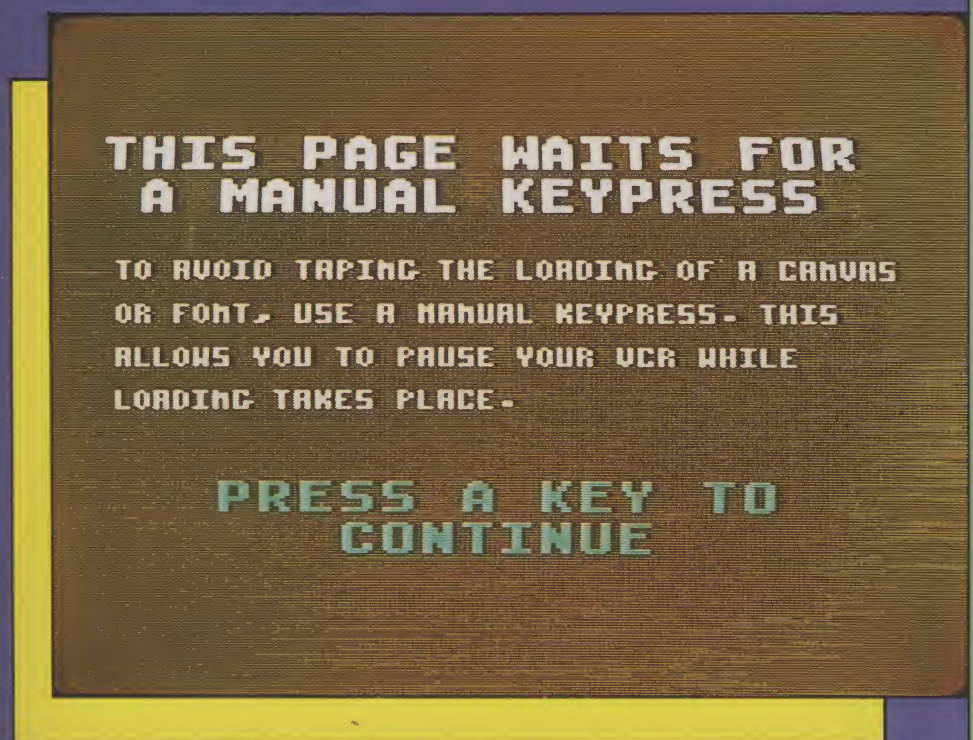
As you prepare each page in your sequence, you can 'play' them back singly, or together in sequence. The process of preparing a page involves selecting your canvas (if any), creating your objects and selecting effects to operate on your page as a whole, or on one or more objects.

An object is essentially a text block. The text may be written using one of a selection of fonts which are on disk. One font is always in memory by default whilst one other may be loaded from disk.

To give you some idea of what effects are available, here is a list of them:-

SCROLL: One or more objects may be scrolled across the screen. The speed and direction of scroll are user-controlled. Several objects may be

TLE SHOP



moved on one page at once. So you could, for example, start with a word in the centre of the screen and 'explode' it using several identical objects or several objects as parts of the original one. Objects may be scrolled completely off the screen since the page size is larger than the video screen.

CYCLE: This effect switches between two adjacent pages at a pre-programmed speed, a preset number of times.

WIPE: This gives a 'tearing' effect from one page to another.

FIZZLE: (My own personal favourite) One page breaks up to reveal the next. The effect is like one picture dissolving to reveal another.

PAINT: Specifically used on objects on a page. There are two types; Paint by Brush, paints the object's letters on the screen whilst Paint by Letter pops the letters in the object onto the screen one by one.

FADE: The page dims out to a black screen.

When preparing your pages, there is a good selection of editing facilities available to you for the manipulation of both pages and objects.

The *Micro Painter Plus* package is used to prepare borders and backgrounds for VTS sequences. Although it is designed for this purpose, I see no reason why it couldn't be used to prepare graphics for other uses, although I don't know if the canvases can be loaded as straight graphic files.

MPP provides you with a number of facilities such as line and shape drawing, fill/paint and zoom. You can fill shapes with patterns which may themselves be edited.

The *Video Title Shop* is an excellent piece of software which has been carefully and professionally prepared. The manual is good, if a little 'American' (in other words, how shall I put it, over-enthusiastic) and covers everything from using the software, to setting up and using video recorders. If you want something to inject extra pazzaz into your sister's wedding video or create your own silent movie (with the fancy dialogue screens), then the *Video Title Shop* is for you!

AT A GLANCE

Title: Video Title Shop
Suppliers: Financial Systems Software
Price: £19.95

WHICH WORD PROC

We put 5 popular packages through the mill

By Tony Hetherington

Buyers Guide to Word Processing

In almost every survey of computer applications word processing dominates the top position because it can even allow a two finger typist to produce perfect documents every time. However, you can save yourself a lot of time, money and trouble by choosing the best word processor to suit your needs.

All of the packages included in this guide are disk based and assume a minimum hardware configuration of C64, disk drive and Commodore or compatible printer. Although there are still some tape based systems to be found for the truly adventurous, most of us would rather do without the tedium of long loading times and the uncertainty of cassette reliability.

As with most areas of computing word processing has its fair share of buzzwords and one you'll hear a lot of is WYSIWYG which stands for **What You See Is What You Get** which loosely translated means that there won't be too many surprises between what you see on the screen and what appears on the paper. Very few C64 word processors have this feature except in page preview form but through a selection of fonts and point sizes your text may actually be WYGIBTWYS (What You Get Is Better Than What You See) which just shows how easy it is to make up computer jargon.

Your choice of word processor may also be affected by what your needs are as these can vary from writing long articles or books to simple letters. As in most things, there seems to be a trade off between the more powerful features and the graphic ones such as fonts and inclusion of clip-art that are in vogue at the moment. So it's up to you whether you want fancy text or a spell checker but here's a brief list of some of the features to look out for.

To be described as user friendly the program should contain adequate editing features such as search and replace and find options as well as the facility to cut, paste, move and copy blocks of text.

If fonts are important to you then check how many are included, the point sizes they are supplied in and how many you can use at a given time.

Spell checkers are very useful particularly those that allow the creation of user dictionaries that you can quickly build up for your own specific needs. Ones that suggest alternatives can be great fun as I once reviewed *Wasteland* and had suggested as an alternative for Wardroids the word wardrobe. Beware, those killer wardrobes!

A Thesaurus may be important to you if you do a lot of writing and want to avoid using the same word twice as after a whirr from the disk drive a series of alternatives will be displayed.

Finally, many packages tout the overrated mailmerge program that will fill in the blanks left on form letters. If you want one of these check where the program can access the list of names or addresses, is it from a word processor file or can it access your database?

To help choose the best word processor for you we have included in this guide reviews of five very different programs and a comparison table so you can check out each program's pros and cons.

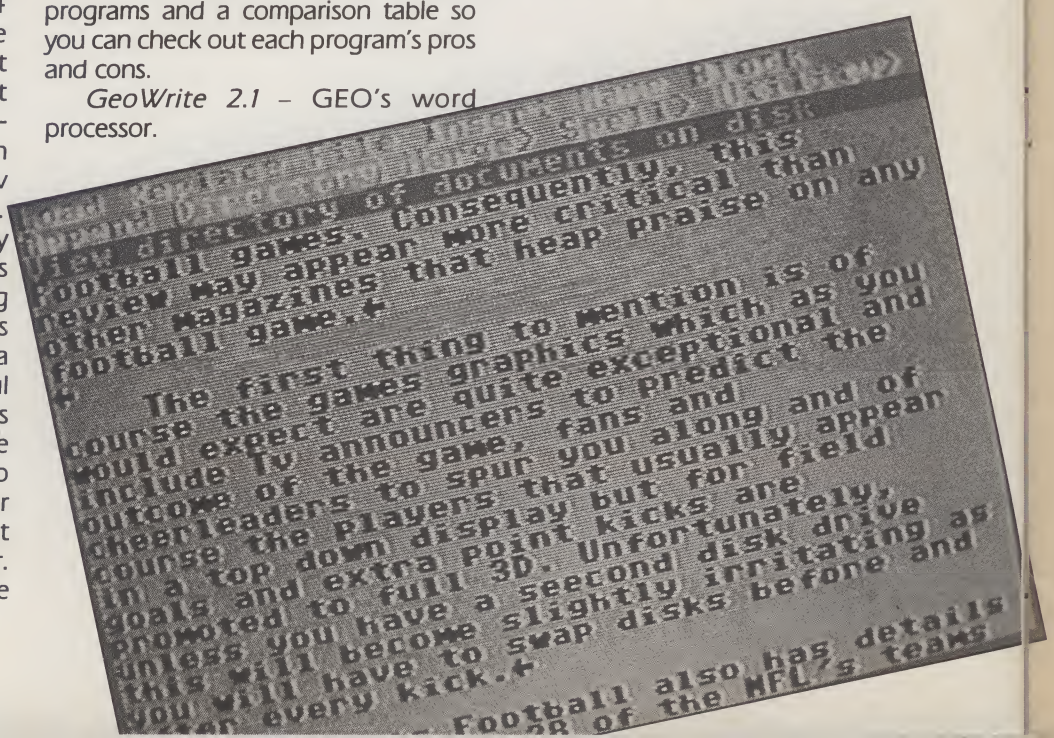
GeoWrite 2.1 – GEO's word processor.

The GEOS word processor has come a long way from the basic text editor that was included with the first edition of GEOS. Now we're up to GEOS *2.0* and *GeoWrite 2.1* is included in the pack (or available separately in the Writer's workshop). Naturally, you need GEOS to run it with all its pros and cons but it does give you the opportunity to build up an integrated system.

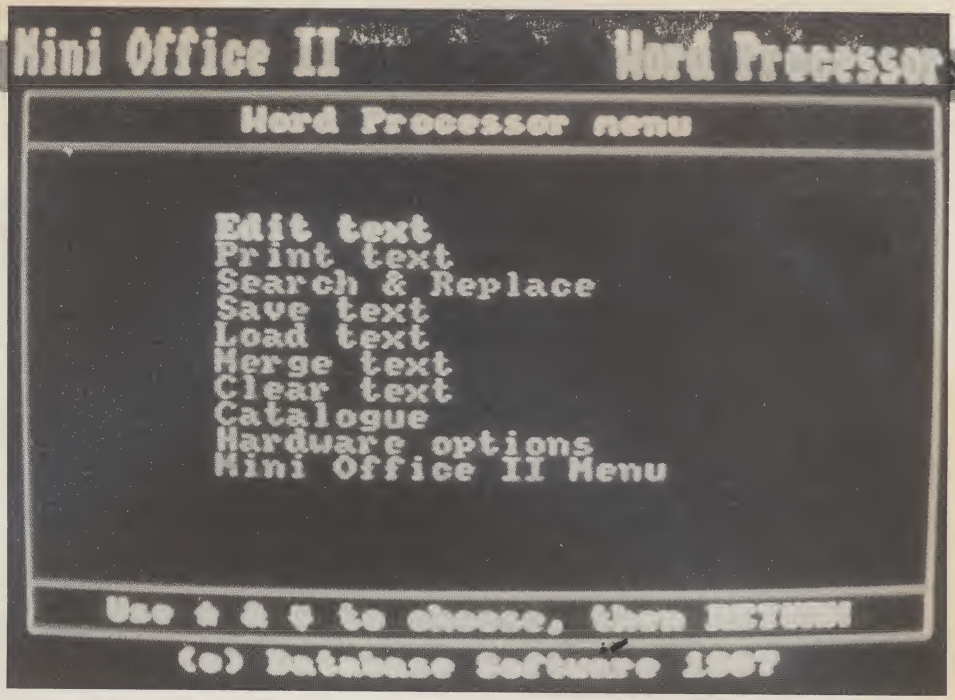
GeoWrite is the wordprocessor if you want to use a variety of fonts as the program includes 10 in seven different styles and more are available via the extra *Font Pack Plus* that adds a further 53 plus a font editor to create your own. You're also free to mix text and graphics on your page that have been created by the companion paint program *GeoPaint* (also included in GEOS) so you can easily include graphs, charts and illustrations in your documents.

GeoWrite is a fully featured word processor offering four types of justification, the ability to expand margins up to eight inches, include headers and footers, create multiple column pages and add headlines and borders. In fact, it's probably the closest a word processor can get to a DTP system.

GeoSpell is its companion spell checker that works outside *GeoWrite* and checks through *GeoWrite* files.



CESSOR



When it finds an error or a spelling that's not in either its 28,000 word dictionary or any of your 10,000 word dictionaries it will suggest possible alternatives and then either substitute, amend or ignore the word.

Geomerge is the GEOS mailmerge program and it is better than the average as you can use IF – ELSE conditions in your mailmerge files and thereby send appropriate form letters to different people in the same mailmerge run. For example, you could send a demand for payment to one customer and receipt to another.

By far the most impressive weapon in *GeoWrite's* armoury is its text grabber utility that can load into *GeoWrite* any C64 word processor file. This means if you upgrade to GEOS you can still use your existing data files.

Superscript – Word processing work horse.

Superscript is the word processing cousin of **Precision Softwares** famous database *Superbase 64* and is an obvious choice for existing Superbase owners. But what does it offer new users? I'll try and explain.

Superscript is controlled entirely through a system of menus, that unlike *GeoWrite's* pull down menus, are more like a double version of a duckshoot menu that is called up by pressing **F1**. The cursor keys are used to highlight an option which when selected by pressing return pulls up another menu.

These menus lead you to all *Superscript's* options and so after an initial learning period, which is helped by a descriptive line whenever an option is highlighted, the system is quite fast to use. This is speeded up by the way that you can bypass the cursor movement by typing in the first letter of the option you require, which explains why the file management menu is called Document because there is also a Feature menu from which you can select enlarged, underlined and condensed text.

From the document menu you can not only load, save and insert files or blocks of text, view the directory and access a list of disk utilities including format, backup and verify, you can also load in the spellchecker that contains a few useful surprises. Instead of the usual fare of simply checking the spelling, which it does very well, you can also look up individual words in the dictionary, amend and print out the user dictionary and display and print information on your document. Three options are available here, one that prints out the words your document has used in alphabetical order, one that lists the words according to the number of times they are used and a third that displays data based on the word count, word length, sentence length and number of paragraphs. An example display would look like this. Document statistics. Total number of words = 1661 Number of unique words = 716 number of sentences = 100 Number of paragraphs = 65 Average word length = 5.93.

This information could help you discover whether you are reaching your target audience as it uses some of the tests that are used in fully fledged writing analysis programs.

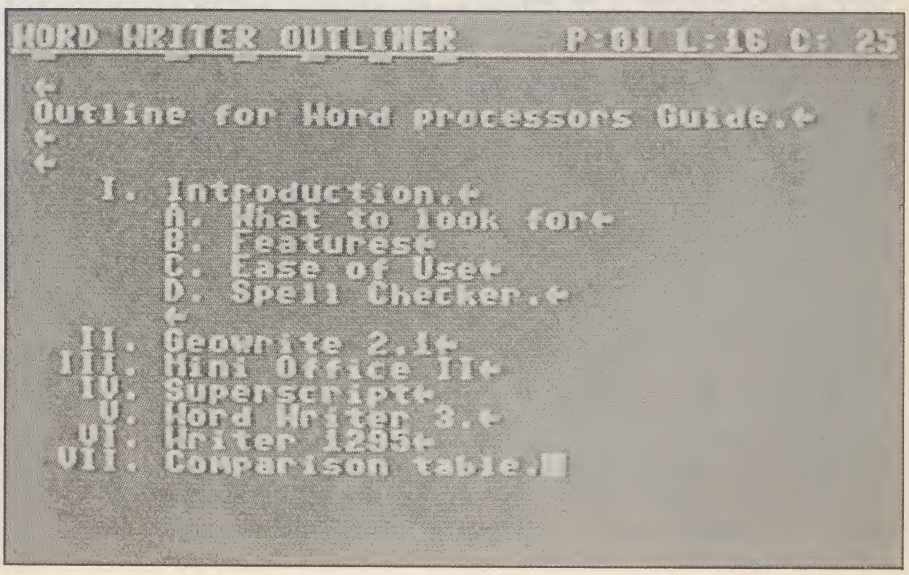
Overall, *Superscript* is a good all round program that offers help to its novices and speed to its practised users.

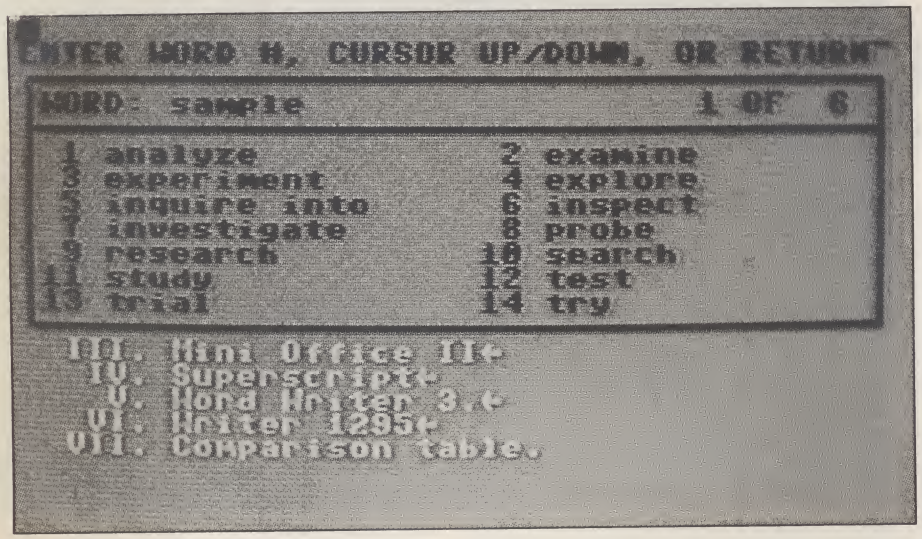
Mini Office I – The integrated package

Mini Office II represents the cheap and cheerful integrated approach to word processing as the pack also includes a limited database, spreadsheet, label printer, comms pack and business graphics utility capable of drawing 2D bar charts, pie charts and line graphs as well as 3D bar charts.

It is also the only system included in this roundup that is also available on cassette as it's the kind of package that it would be fair to say neither part of the pack could stand on its own due to the limited features.

The word processor is selected from the main menu that then displays the word processor menu that has edit





text, print text and search and replace as separate options. The screen display is best described as crude with the initial screen showing the START and END markers, the time, characters remaining (30,490 before you type a word) and a word count. That's the strange thing about choosing a word processor. This one might be the simplest and sparsest included in this guide, it is also the only one that provides a continuous word count display. (Superscript has one but this is only displayed after spellchecking). This is a shame as many users have to write to length and would find this a useful facility.

You enter text in either insert or overwrite mode and can also use limited block control commands to move, copy and delete text. A series of embedded commands are used to set margins, text justification, set tab positions, line spacing, define headers and footers and get a file from the database module to insert a name and address in a letter.

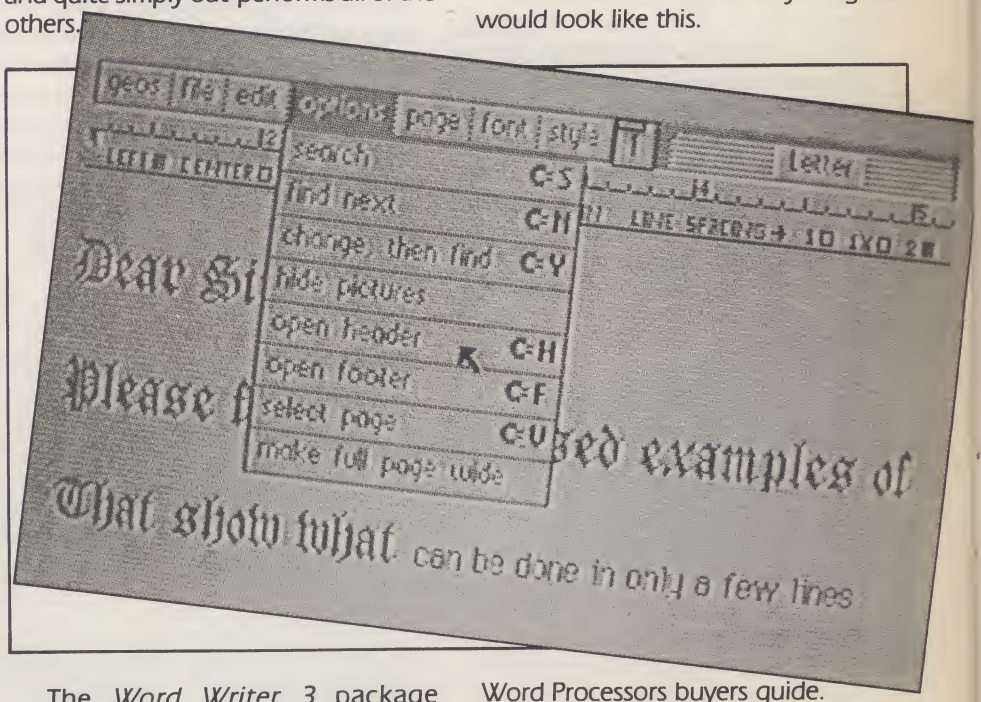
As mentioned earlier Search and Replace is a separate menu option but can search for selected words for what the manual optimistically describes as "indexing" and search for and replace consistent spelling mistakes.

Finally, as an added bonus the package uses its clock and word count feature to measure your typing speed should you need to know this spurious information.

To judge the mini office word processor on its own would be harsh and you'd probably only buy it as part of the full package. As such it represents excellent value for money, for where else could you get a word processor, database, spreadsheet, label printer, graph utility and comms package for only 19.95?

Word Writer 3 – Power program including a spellchecker, thesaurus and outliner.

Word Writer 3 offers more power and features than any other C64 word processing package but without penalising its users with a restrictive price and quite simply out-performs all of the others.



The *Word Writer 3* package includes two disks, a manual and two keyboard overlays that provide an at-a-glance guide to the key presses required for the programs editing, cursor movement and advanced features. These key guides and a series of double keypresses form the basic control mechanism of the program, however when an option is selected a brief three or four item menu of options may appear at the top of the page. The exceptions to this are the spell checker and thesaurus that creeate their own windows full of numbered alternatives. All you have to do is type in the appropriate number.

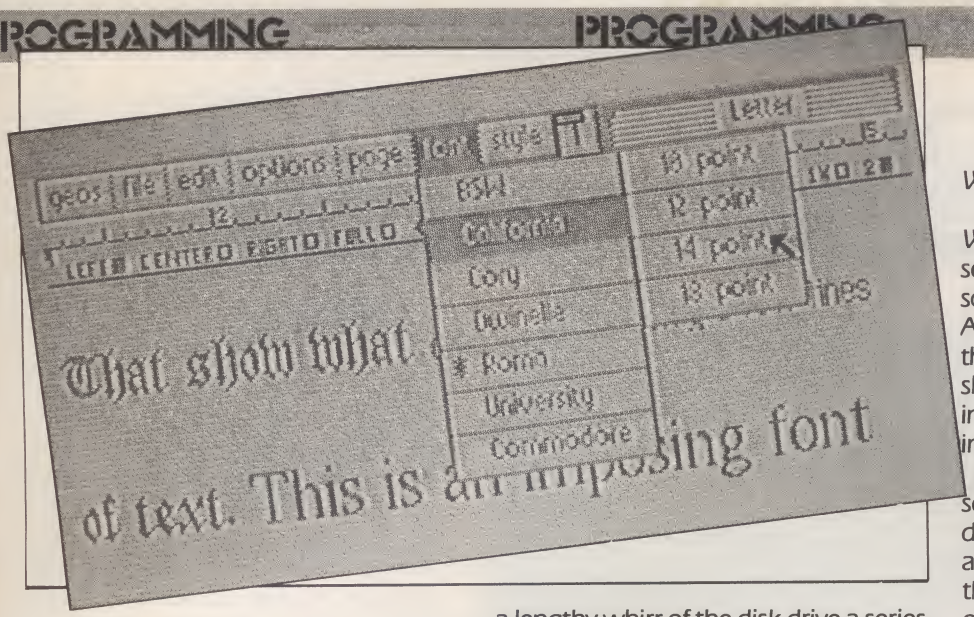
Word Writer 3 boasts an impressive range of features including not only an outliner, spellchecker and thesaurus but also the ability to chain and merge documents together, over 200 tabs for accurate text placement, a competent selection of block control and editing functions and the ability to redefine the screen colours to suit your mood.

The outliner should form the first stage of your document writing process, particularly if you're writing long essays, articles or even a book. It provides a framework with which you can outline your writing and organise your thoughts. It does this through a series of nested levels that start with main topic headlines that are denoted by Roman numerals and then five sub levels represented by capital letters, numbers, small letters and bracketed numbers respectively. For example, part of the outline for this buyers guide would look like this.

Word Processors buyers guide.

I. Intro
  A. Jargon
  B. User friendly
  C. Features

    1. Fonts.



2. Spellcheckers

3. Mailmerge II. GeoWrite
A. Fonts
B. Features.
C. GeoSpell
D. GeoMerge
E. Text Grabber. III. Superscript

The power of the outliner comes into play when you want to alter, add to and edit it and as a single key press will reorganise it. When it's complete and you exit the outliner it appears at the start of your word processor document so you can refer to it as you type the body text.

No matter how good a typist you think you are you will always make spelling mistakes or typing mistakes so it's useful to have a spellchecker included in the package. One that contains a dictionary of 85,000 words is particularly useful and you can also add your own dictionaries to account for your own specific applications and the associated jargon. However, the spell checker can only cope with documents that are up to 10 pages in length so longer ones need to be split up and saved individually. (They can be remerged together with the Appending facility). Spellchecking takes a few minutes as it sorts the words of your documents into alphabetical order and then compares them against its 26 letter dictionairies. Finally, it highlights suspect words and prompts you to ignore, replace or edit them.

A thesaurus should always be considered as a luxury but it's one that's included in this package. Its 65,000 words are stored on the package's second disk and accessed by pressing C= 0. You can either look up the word by the cursor or type in one and after a lengthy whirr of the disk drive a series of alternatives will be displayed. In some cases this may be up to five pages worth and typing in the appropriate letter will substitute the word in your document.

Pressing **F5** punches up an onscreen calculator that may prove invaluable to some users, useful to others and provide a novelty for the rest. It is more than just a gimmick as it can add, subtract, multiply, divide, change sign and raise to a power.

Word Writer 3 can also produce Pet ASCII files for transfer by modem of file to other machines and word processors and is also GEOS compatible and can be loaded from within GEOS by a click of its icon.

You'd have to agree that *Word Writer 3* has an impressive range of features but it still has its limitations. It has no graphics facility so scores poorly on the business or school report side and may cost more than many users may wish to spend. For example, *Superscript* is just under half the price and although *GeoWrite* is the same price the new GEOS 2.0 pack also includes GEOS and geoPaint.

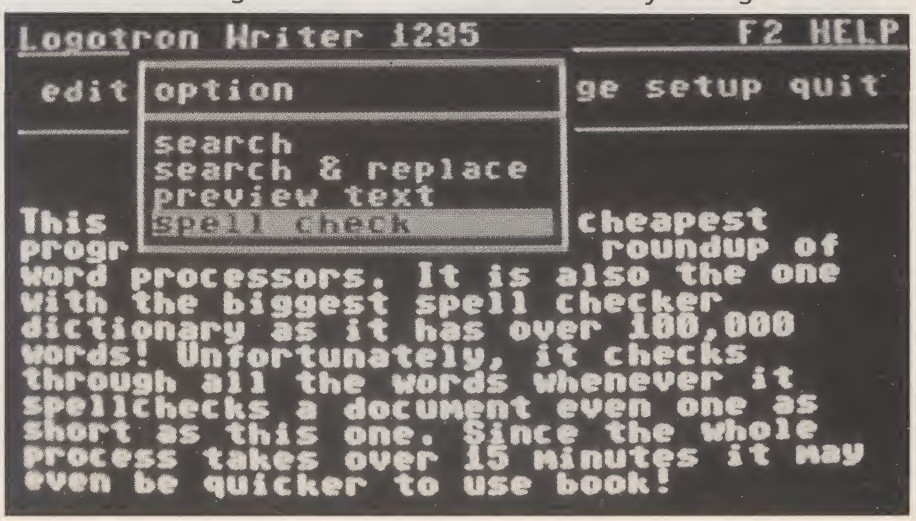
Writer 1295 – Budget word processing.

Writer 1295 is part of **Logotrons 1295** series of programs that offer cut price software without cut down features. As such, it offers a reporting option that can be used to print the spreadsheet figures from Planner 1295 and incorporate the data from Filer 1295 into mail merged letters.

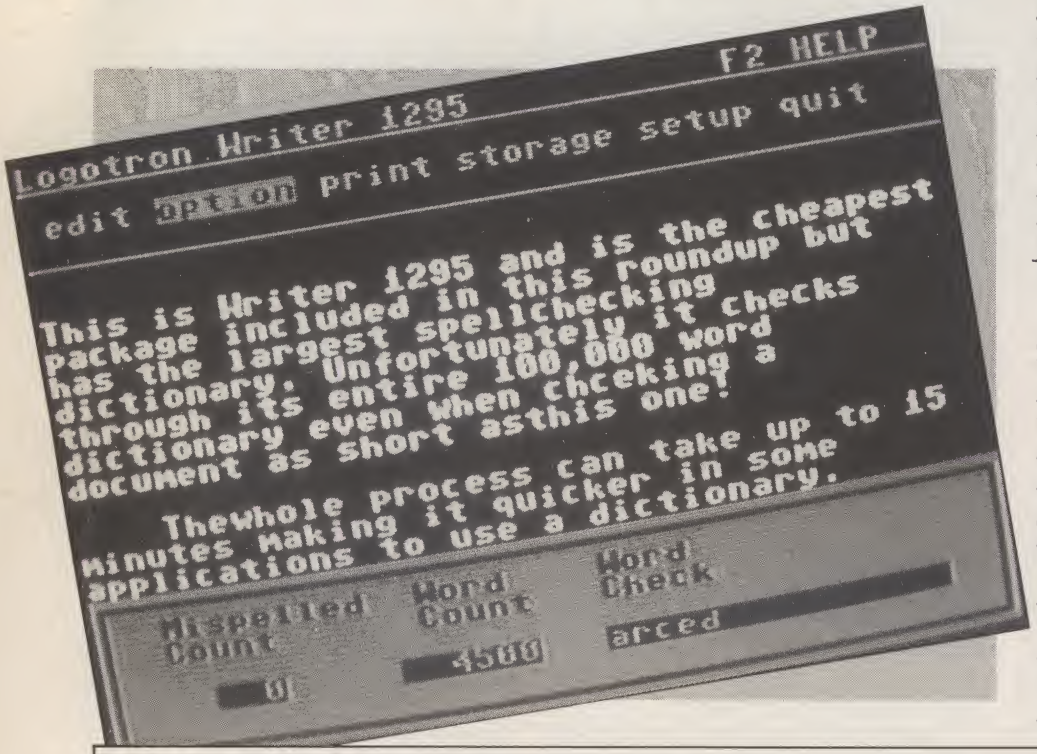
Writer 1295 is operated through a series of pull down menus that are displayed by pressing the STOP key and accessed with the cursor key. Through these options you can access files held on disk, set up the hardware you will use (printer, second disk drive etc), define the screens colours you will use, edit a page, search and search and replace (again in a separate menu!), edit pages of text and spellcheck and print the final result.

While creating and editing a page of text the cursor is always in insert mode which works well for most things with the possible exception of creating tables. It's the system used by the *Amstrad PCW* and is a lot better than *Superscript's* overwrite style where to insert a word you must first insert the spaces it will need.

The big surprise about *Writer 1295* is its 100,000 word spellchecker that offers a larger dictionary than all the other word processors in this guide. So what's the catch? I hear you ask. The answer is that while there is a full spellchecker it is not as efficiently organised as the others and takes a lot longer even to check a small document. Unlike the others that sort the documents words and then look them up in their dictionaries this one does the reverse and flips through its entire dictionary looking for words in



our document! When it has completed this mammoth task it displays the suspect words and allows you the option to accept or edit them. However, now the cursor is in overwrite mode as opposed to the insert mode



throughout document creation.

Any disk based word processor costing only £12.95 that integrates with a spreadsheet and database and offers features such as a 100,000 word spellchecker should be regarded as a gem. Unfortunately, its a flawed gem as it has some curious key presses and a quite ludicrous method of checking spelling. To see what I mean try looking at every word in the dictionary in turn just to find a word you want to look up!

The following table has been prepared to help you choose the best word processor for your needs. Consequently it's more comprehensive than the usual table you find and includes a list of major features followed by a set of ratings that judge each word processor against a variety of criteria and the most popular applications. These include short applications such as a letter or a memo, a business report including charts and figures and a book or other long feature or article. Finally there are details of supplier, price and a contact number.

	GeoWrite 2.1	Super-script	Word Writer 3	Mini Office II	Writer 1295
Spellchecker	Y	Y	Y	N	Y
dict size	28,000	60,000	85,000	0	100,000
User dict.	Y	Y	Y	N	N
Thesaurus	N	N	Y	N	N
Outliner	N	N	Y	N	N
No. Fonts	10	1	1	1	1
Pulldown menus	Y	N	N	N	Y
kbrd shortcuts	Y	Y	Y	Y	Y
Word Count	N	Y	N	N	N
Headers/footers	Y	Y	Y	Y	Y
Search/replace	Y	Y	Y	Y	Y
Graphics?	Y	N	N	N	N
ASCII?	N	Y	Y	Y	N
Ratings %					
Ease of use	90%	80%	70%	60%	65%
Power	75%	78%	95%	50%	50%
features	70%	80%	90%	40%	55%
editing	65%	75%	70%	40%	60%
value	70%	90%	90%	75%	80%
Applications %					
Letter	35%	75%	70%	60%	40%
Report	90%	30%	30%	40%	25%
Book	30%	80%	90%	20%	40%
Supplier	F.S.S.L	Precision	F.S.S.L.	Database	Logotron
Contact No.	(0386) 553153	01-330 7166	(0386) 553153	(0625) 878888	(0233) 323656
Price	39.95	24.95	39.95	19.95	12.95

COMMODORE Disk User

*delivered to your door FREE !**

That's right, if you take out a year's subscription to Commodore Disk User we will make sure that it is delivered to your door each issue at no extra charge*. Just fill in the coupon below and send it to the address given with a cheque, money order or credit card instructions to cover the cost of the subscription. We'll do the rest.

*** Overseas subscription rates include postage.**

UK:	£16.50
EUROPE:	£19.50
MIDDLE EAST:	£19.65
FAR EAST:	£20.80
REST OF THE WORLD:	£19.90
or USA:	$35.00

Airmail Rates on Request.

Send this form with your remittance to:
INFONET LTD.
5 River Park Estate,
Billet Lane,
BERKHAMSTED,
Herts. HP4 1HL,
United Kingdom

Please commence my subscription to Commodore Disk User with the ..... issue. I enclose my cheque/money order for £..... made payable to ARGUS SPECIALIST PUBLICATIONS

Or Debit my Access/Visa number ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Valid from ..... to ..... Signature .....

Name ..... Address .....

..... Postcode .....

☐ New subscriber ☐ Subscription renewal/extension (delete as applicable)


AMIGA
ATARI ST
IBM PC
APPLE MAC
ACORN
COMMODORE 64
AMSTRAD CPC
ZX SPECTRUM
MICRO Music
THE SPECIALIST MUSIC MAGAZINE FOR COMPUTER USERS
SEPTEMBER 1989 £1.75
USIVE
SIMON HARRIS TERVIEW
MM US SHOW NEPORT
HARNESS THE W
KAWAI K1-R AND EDITOR SOFT
REV
ISSN 0956-0874
08
9 770956 087004
THE TIME HAS COME . . .
Britain's First Music Magazine for the Computer User!
Featuring Reviews on Hardware 5 Software across all formats — Micro Music is the magazine the Computer Musician has been waiting for.
Available 14th July from all good Newsagents